

(19)



JAPANESE PATENT OFFICE

PATENT ABSTRACTS OF JAPAN

(11) Publication number: 10097483 A

(43) Date of publication of application: 14 . 04 . 98

(51) Int. Cl

G06F 13/00

(21) Application number: 09141762

(71) Applicant: INTERNATL BUSINESS MACH CORP <IBM>

(22) Date of filing: 30 . 05 . 97

(72) Inventor: IYENGAR ARUN

(30) Priority: 07 . 06 . 96 US 96 660633

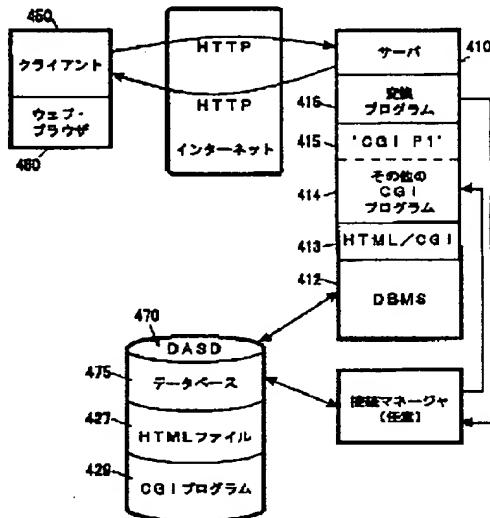
(54) STATE STORING METHOD AND DEVICE IN
NON-STATE NETWORK PROTOCOL

COPYRIGHT: (C)1998,JPO

(57) Abstract:

PROBLEM TO BE SOLVED: To store state information for a duration of conversation and to provide it for entire services by identifying all of the links which are included in an output from a service, recursively incorporating state information into identified links and communicating an output to a client in response to them.

SOLUTION: A client 450 accesses a home page which exists on a server 410'. starts to browse a product catalog and continues to browse a catalog merchandise by selecting a hyperlink from an online product catalog. When the client 450 accesses a system for the first time, he selects a user's ID and a password and offers some additional and optional information to the server 410'. A switching program 416 incorporates the user's ID and a session ID into a conversation, while a user shows an additional product, adds an additional item to a purchase list and commits the purchase. Then he can show and update database information 475, and the state information is stored.



特開平10-97483

(43) 公開日 平成10年(1998)4月14日

(51) Int. C1. 6
G 06 F 13/00 357F I
G 06 F 13/00 357 Z

審査請求 未請求 請求項の数 50 O L

(全25頁)

(21) 出願番号 特願平9-141762
 (22) 出願日 平成9年(1997)5月30日
 (31) 優先権主張番号 08/660633
 (32) 優先日 1996年6月7日
 (33) 優先権主張国 米国 (U.S.)

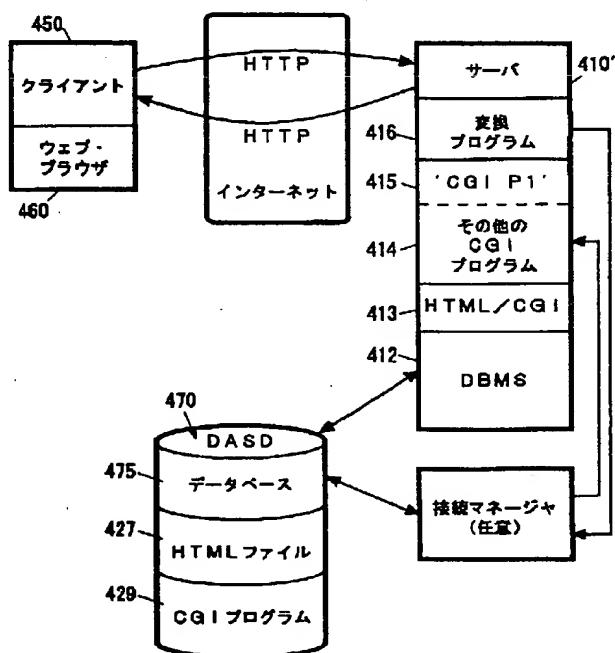
(71) 出願人 390009531
 インターナショナル・ビジネス・マシーンズ・コーポレイション
 INTERNATIONAL BUSINESS MACHINES CORPORATION
 アメリカ合衆国10504、ニューヨーク州
 アーモンク (番地なし)
 (72) 発明者 アルン・イエンガー
 アメリカ合衆国10598、ニューヨーク州ヨ
 ークタウン・ハイツ パーク・レーン
 1160
 (74) 代理人 弁理士 坂口 博 (外1名)

(54) 【発明の名称】無状態ネットワーク・プロトコルにおける状態の保存方法および装置

(57) 【要約】

【課題】 HTTPなどの無状態プロトコルを使用してワールド・ワイド・ウェブ(WWW)などのネットワーク上で通信するコンピュータにおいて状態を保存するための方法およびシステムを提供する。

【解決手段】 サービスを実行し、サービスからの出力に含まれるすべての連結(ハイパーリンク)を識別し、クライアントに送られる出力ですべての識別済み連結に再帰的に状態情報を組み込むことにより、サーバからサービスを要求するクライアントとの間の会話で状態が保存される。状態情報の組込みは、サーバによって実行し、サーバからクライアントに通信することができる。あるいは、動的にダウンロード可能なプログラム・コードを使用して、クライアント側で状態情報を組み込むこともできる。追加の機能により、所定の基準に応じてサービスからのデータ出力とハイパーリンクのフィルタ処理または追加が可能になる。



【特許請求の範囲】

【請求項1】無状態プロトコルによりクライアントに対してネットワーク化された1つまたは複数のサーバからサービスを要求できるようになっているクライアントとの間の会話で状態情報を保存するためのコンピュータ化した方法において、前記サービスがクライアントが要求可能なデータおよびプログラムのうちの1つまたは複数を含み、会話は前記サービスに関するクライアントと1つまたは複数のサーバとの間の一連の通信であり、サーバからの各応答は前記サービスに関する別の要求を可能にする1つまたは複数の連結を含み、クライアントは会話を続行するためにその連結のうちの1つを呼び出さなければならず、前記方法が、

クライアントが無状態プロトコルを使用してサーバとの会話を開始するステップと、

サービス要求が状態情報の保存を必要とする時期を検出するステップと、

前記検出ステップに応答して、前記サービスを実行し、前記サービスからの出力に含まれるすべての連結を識別するステップと、

すべての識別済み連結に再帰的に状態情報を組み込むステップと、

前記組込みステップに応答して、出力をクライアントに通信するステップとを含み、会話の持続期間中に状態情報が保存され、すべてのサービスに提供されることを特徴とする方法。

【請求項2】前記組込みステップがサーバによって実行され、前記通信ステップが前記組込みステップに応答して行われることを特徴とする、請求項1に記載の方法。

【請求項3】サーバに結合されたメモリに状態情報の少なくとも一部を格納するステップをさらに含み、前記組込みステップが前記すべての識別済み連結に状態情報の前記一部を表すインデックスを組み込むステップを含むことを特徴とする、請求項2に記載の方法。

【請求項4】出力をクライアントに通信する前記ステップに応答して前記組込みステップを実行するために、コンピュータ・プログラム・コードをクライアントに動的にダウンロードするステップをさらに含むことを特徴とする、請求項1に記載の方法。

【請求項5】クライアントに結合されたメモリに状態情報の少なくとも一部を格納するステップをさらに含み、前記組込みステップが状態情報の前記一部を表すインデックスを組み込むステップを含むことを特徴とする、請求項4に記載の方法。

【請求項6】クライアントが、前記すべての識別済み連結から状態情報が組み込まれた第2の連結を選択するステップと、

前記第2の連結から状態情報を復元し、復元した状態情報により関連の第2のサービスを呼び出すステップと、前記第2のサービスからの出力に関連するすべての連結

において再帰的に状態情報を識別し、組み込むステップとをさらに含むことを特徴とする、請求項1に記載の方法。

【請求項7】状態情報と特定の会話を相関させるステップをさらに含むことを特徴とする、請求項1に記載の方法。

【請求項8】クライアントとサーバがワールド・ワイド・ウェブを介してネットワーク化され、無状態プロトコルがハイパーテキスト転送プロトコルであり、連結が、

10 ハイパーテキスト・マークアップ言語ファイルおよび共通ゲートウェイ・インターフェース・プログラムの1つへのハイパーリンクであることを特徴とする、請求項1に記載の方法。

【請求項9】所定の基準に応じて前記サービスからのデータ出力と前記ハイパーリンクの1つをフィルタ処理するステップをさらに含むことを特徴とする、請求項8に記載の方法。

【請求項10】所定の基準に応じて前記サービスからの前記出力にデータと前記ハイパーリンクの1つを追加するステップをさらに含むことを特徴とする、請求項8に記載の方法。

【請求項11】前記組込みステップが、識別済み連結と引数として渡された状態情報によりCGI変換プログラムを呼び出すためにHTMLファイルを求める要求である識別済み連結を変更するステップをさらに含むことを特徴とする、請求項8に記載の方法。

【請求項12】前記組込みステップが、識別済み連結と引数として渡された状態情報によるCGIプログラムへの呼出しである識別済み連結を変更するステップをさらに含み、前記組込みステップがCGIプログラムによって実行されることを特徴とする、請求項8に記載の方法。

【請求項13】前記組込みステップが、識別済み連結によりCGI変換プログラムを呼び出すためのCGIプログラムへの呼出しである識別済み連結と、CGIプログラムに関連する引数の数を示す引数カウンタと、引数として渡された状態情報とを変更するステップをさらに含み、前記組込みステップが変換プログラムによって実行されることを特徴とする、請求項8に記載の方法。

【請求項14】無状態プロトコルによりクライアントに対してネットワーク化された1つまたは複数のサーバからサービスを要求できるようになっているクライアントとの間の会話で状態情報を保存するための方法を提供するためにコンピュータによって実行可能な複数の命令からなるプログラムを具体的に実施するコンピュータによって読み取り可能なプログラム記憶装置において、前記サービスがクライアントが要求可能なデータおよびプログラムのうちの1つまたは複数を含み、会話は前記サービスに関するクライアントと1つまたは複数のサーバとの

間の一連の通信であり、サーバからの各応答は前記サービスに関する別の要求を可能にする1つまたは複数の連結を含み、クライアントは会話を続行するためにその連結のうちの1つを呼び出さなければならず、前記方法が、

クライアントが無状態プロトコルを使用してサーバとの会話を開始するステップと、

サービス要求が状態情報の保存を必要とする時期を検出するステップと、

前記検出ステップに応答して、前記サービスを実行し、前記サービスからの出力に含まれるすべてのハイパリンクを識別するステップと、

すべての識別済みハイパーリンクに再帰的に状態情報を組み込むステップと、

前記組込みステップに応答して、出力をクライアントに通信するステップとを含み、会話の持続期間中に状態情報が保存され、すべてのサービスに提供されることを特徴とするプログラム記憶装置。

【請求項15】前記組込みステップがサーバによって実行され、前記通信ステップが前記組込みステップに応答して行われることを特徴とする、請求項14に記載の方法ステップを実行するためにコンピュータによって実行可能な複数の命令からなるプログラムを具体的に実施するコンピュータによって読み取り可能なプログラム記憶装置。

【請求項16】サーバに結合されたメモリに状態情報の少なくとも一部を格納するステップをさらに含み、前記組込みステップが前記すべての識別済み連結に状態情報の前記一部を表すインデックスを組み込むステップを含むことを特徴とする、請求項15に記載の方法ステップを実行するためにコンピュータによって実行可能な複数の命令からなるプログラムを具体的に実施するコンピュータによって読み取り可能なプログラム記憶装置。

【請求項17】出力をクライアントに通信する前記ステップに応答して前記組込みステップを実行するために、コンピュータ・プログラム・コードをクライアントに動的にダウンロードするステップをさらに含むことを特徴とする、請求項14に記載の方法ステップを実行するためにコンピュータによって実行可能な複数の命令からなるプログラムを具体的に実施するコンピュータによって読み取り可能なプログラム記憶装置。

【請求項18】クライアントに結合されたメモリに状態情報の少なくとも一部を格納するステップをさらに含み、前記組込みステップが状態情報の前記一部を表すインデックスを組み込むステップを含むことを特徴とする、請求項14に記載の方法ステップを実行するためにコンピュータによって実行可能な複数の命令からなるプログラムを具体的に実施するコンピュータによって読み取り可能なプログラム記憶装置。

【請求項19】クライアントが、前記すべての識別済み

連結から状態情報が組み込まれた第2の連結を選択するステップと、

前記第2の連結から状態情報を復元し、復元した状態情報により関連の第2のサービスを呼び出すステップと、前記第2のサービスからの出力に関連するすべての連結において再帰的に状態情報を識別し、組み込むステップとをさらに含むことを特徴とする、請求項14に記載の方法ステップを実行するためにコンピュータによって実行可能な複数の命令からなるプログラムを具体的に実施するコンピュータによって読み取り可能なプログラム記憶装置。

【請求項20】状態情報と特定の会話を相関させるステップをさらに含むことを特徴とする、請求項14に記載の方法ステップを実行するためにコンピュータによって実行可能な複数の命令からなるプログラムを具体的に実施するコンピュータによって読み取り可能なプログラム記憶装置。

【請求項21】クライアントとサーバがワールド・ワイド・ウェブを介してネットワーク化され、無状態プロトコルがハイパーテキスト転送プロトコルであり、連結が、ハイパーテキスト・マークアップ言語ファイルおよび共通ゲートウェイ・インターフェース・プログラムの1つへのハイパーリンクであることを特徴とする、請求項14に記載の方法ステップを実行するためにコンピュータによって実行可能な複数の命令からなるプログラムを具体的に実施するコンピュータによって読み取り可能なプログラム記憶装置。

【請求項22】所定の基準に応じて前記サービスからのデータ出力と前記ハイパーリンクの1つをフィルタ処理するステップをさらに含むことを特徴とする、請求項21に記載の方法ステップを実行するためにコンピュータによって実行可能な複数の命令からなるプログラムを具体的に実施するコンピュータによって読み取り可能なプログラム記憶装置。

【請求項23】所定の基準に応じて前記サービスからの前記出力にデータと前記ハイパーリンクの1つを追加するステップをさらに含むことを特徴とする、請求項21に記載の方法ステップを実行するためにコンピュータによって実行可能な複数の命令からなるプログラムを具体的に実施するコンピュータによって読み取り可能なプログラム記憶装置。

【請求項24】前記組込みステップが、識別済み連結と引数として渡された状態情報によりCGI変換プログラムを呼び出すためにHTMLファイルを求める要求である識別済み連結を変更するステップをさらに含むことを特徴とする、請求項21に記載の方法ステップを実行するためにコンピュータによって実行可能な複数の命令からなるプログラムを具体的に実施するコンピュータによって読み取り可能なプログラム記憶装置。

【請求項25】前記組込みステップが、

識別済み連結と引数として渡された状態情報によるCGIプログラムへの呼出しである識別済み連結を変更するステップをさらに含み、前記組込みステップがCGIプログラムによって実行されることを特徴とする、請求項21に記載の方法ステップを実行するためにコンピュータによって実行可能な複数の命令からなるプログラムを具体的に実施するコンピュータによって読み取り可能なプログラム記憶装置。

【請求項26】前記組込みステップが、
識別済み連結によりCGI変換プログラムを呼び出すためのCGIプログラムへの呼出しである識別済み連結と、CGIプログラムに関連する引数の数を示す引数カウンタと、引数として渡された状態情報とを変更するステップをさらに含み、前記組込みステップが変換プログラムによって実行されることを特徴とする、請求項21に記載の方法ステップを実行するためにコンピュータによって実行可能な複数の命令からなるプログラムを具体的に実施するコンピュータによって読み取り可能なプログラム記憶装置。

【請求項27】無状態プロトコルによりクライアントに対してネットワーク化された1つまたは複数のサーバからサービスを要求できるようになっているクライアントとの間の会話で状態情報を保存するためのコンピュータ・システムにおいて、前記サービスがクライアントが要求可能なデータおよびプログラムのうちの1つまたは複数を含み、会話は前記サービスに関するクライアントと1つまたは複数のサーバとの間の一連の通信であり、サーバからの各応答は前記サービスに関する別の要求を可能にする1つまたは複数の連結を含み、クライアントは会話を続行するためにその連結のうちの1つを呼び出さなければならず、前記システムが、

無状態プロトコルを使用してサーバとの会話を開始できるようになっているクライアントと、

サービス要求が状態情報の保存を必要とする時期を検出する状態検出論理と、

前記検出ステップに応答して、前記サービスからの出力に含まれるすべての連結を識別する探索論理と、

すべての識別済み連結に再帰的に状態情報を組み込む変換プログラム論理と、

出力をクライアントに通信する通信論理とを含み、会話の持続期間中に状態情報が保存され、すべてのサービスに提供されることを特徴とするコンピュータ・システム。

【請求項28】前記変換プログラム論理がサーバによって実行され、前記通信論理により状態情報が組み込まれた出力がサーバからクライアントに通信されることを特徴とする、請求項27に記載のコンピュータ・システム。

【請求項29】状態情報の少なくとも一部を格納するためにサーバに結合されたメモリをさらに含み、前記変換

プログラム論理が前記すべての識別済み連結に状態情報の前記一部を表すインデックスを組み込めるようになっていることを特徴とする、請求項28に記載のコンピュータ・システム。

【請求項30】前記通信論理により状態情報が組み込まれていない出力がサーバからクライアントに通信され、サーバが実行のために前記変換プログラム論理をクライアントに動的にダウンロードできるようになっていることを特徴とする、請求項27に記載のコンピュータ・システム。

【請求項31】状態情報の少なくとも一部を格納するためにクライアントに結合されたメモリをさらに含み、前記変換プログラム論理がさらに状態情報の前記一部を表すインデックスを組み込めるようになっていることを特徴とする、請求項30に記載のコンピュータ・システム。

【請求項32】クライアントが、前記すべての識別済み連結から状態情報が組み込まれた第2の連結を選択し、変換プログラム論理が、さらに前記第2の連結から状態

20 情報を復元し、復元した状態情報により関連の第2のサービスを呼び出し、前記第2のサービスからの出力に関連するすべての連結において再帰的に状態情報を識別し、組み込めるようになっていることをさらに含むことを特徴とする、請求項27に記載のコンピュータ・システム。

【請求項33】状態情報と特定の会話とが関連することを特徴とする、請求項27に記載のコンピュータ・システム。

【請求項34】クライアントとサーバがワールド・ワイド・ウェブを介してネットワーク化され、無状態プロトコルがハイパーテキスト転送プロトコルであり、連結が、ハイパーテキスト・マークアップ言語ファイルおよび共通ゲートウェイ・インターフェース・プログラムの1つへのハイパーリンクであることを特徴とする、請求項27に記載のコンピュータ・システム。

【請求項35】所定の基準に応じて前記サービスからのデータ出力と前記ハイパーリンクの1つをフィルタ処理するフィルタ論理をさらに含むことを特徴とする、請求項34に記載のコンピュータ・システム。

40 【請求項36】所定の基準に応じて前記サービスからの前記出力にデータと前記ハイパーリンクの1つを追加する統合論理をさらに含むことを特徴とする、請求項34に記載のコンピュータ・システム。

【請求項37】前記組込みステップが、識別済み連結と引数として渡された状態情報によりCGI変換プログラムを呼び出すためにHTMLファイルを求める要求である識別済み連結を変更するステップをさらに含むことを特徴とする、請求項34に記載のコンピュータ・システム。

50 【請求項38】前記組込みステップが、

識別済み連結と引数として渡された状態情報によるCG Iプログラムへの呼出しである識別済み連結を変更するステップをさらに含み、前記組込みステップがCG Iプログラムによって実行されることを特徴とする、請求項34に記載のコンピュータ・システム。

【請求項39】前記組込みステップが、

識別済み連結によりCG I変換プログラムを呼び出すためのCG Iプログラムへの呼出しである識別済み連結と、CG Iプログラムに関連する引数の数を示す引数カウンタと、引数として渡された状態情報とを変更するステップをさらに含み、前記組込みステップが変換プログラムによって実行されることを特徴とする、請求項34に記載のコンピュータ・システム。

【請求項40】無状態プロトコルによりクライアントに対してネットワーク化された1つまたは複数のサーバからサービスを要求できるようになっているクライアントとの間の会話で状態情報を保存するためのコンピュータ・システムにおいて、前記サービスがクライアントが要求可能なデータおよびプログラムのうちの1つまたは複数を含み、会話は前記サービスに関するクライアントと1つまたは複数のサーバとの間の一連の通信であり、サーバからの各応答は前記サービスに関する別の要求を可能にする1つまたは複数の連結を含み、クライアントは会話を続行するためにその連結のうちの1つを呼び出さなければならず、前記システムが、

無状態プロトコルを使用してサーバとの会話を開始できるようになっているクライアントと、
サービス要求が状態情報の保存を必要とする時期を検出する状態検出手段と、

前記検出手段に応答して、前記サービスからの出力に含まれるすべての連結を識別する探索手段と、すべての識別済み連結に再帰的に状態情報を組み込む変換プログラム手段と、

出力をクライアントに通信する通信手段とを含み、会話の持続期間中に状態情報が保存され、すべてのサービスに提供されることを特徴とするコンピュータ・システム。

【請求項41】前記変換プログラム手段がサーバによって実行され、前記通信手段により状態情報が組み込まれた出力がサーバからクライアントに通信されることを特徴とする、請求項40に記載のコンピュータ・システム。

【請求項42】状態情報の少なくとも一部を格納するためにサーバに結合されたメモリをさらに含み、前記変換プログラム手段が前記すべての識別済み連結に状態情報の前記一部を表すインデックスを組み込めるようになっていることを特徴とする、請求項41に記載のコンピュータ・システム。

【請求項43】前記通信手段により状態情報が組み込まれていない出力がサーバからクライアントに通信され、

サーバが実行のために前記変換プログラム手段をクライアントに動的にダウンロードできるようになっていることを特徴とする、請求項40に記載のコンピュータ・システム。

【請求項44】状態情報の少なくとも一部を格納するためにクライアントに結合されたメモリをさらに含み、前記変換プログラム手段がさらに状態情報の前記一部を表すインデックスを組み込めるようになっていることを特徴とする、請求項43に記載のコンピュータ・システム。

【請求項45】クライアントが、前記すべての識別済み連結から状態情報が組み込まれた第2の連結を選択し、変換プログラム手段が、さらに前記第2の連結から状態情報を復元し、復元した状態情報により関連の第2のサービスを呼び出し、前記第2のサービスからの出力に関連するすべての連結において再帰的に状態情報を識別し、組み込めるようになっていることをさらに含むことを特徴とする、請求項41に記載のコンピュータ・システム。

【請求項46】所定の基準に応じて前記サービスからの前記出力にデータと前記ハイパーリンクの1つを追加する統合手段をさらに含むことを特徴とする、請求項45に記載のコンピュータ・システム。

【請求項47】クライアントとサーバがワールド・ワイド・ウェブを介してネットワーク化され、無状態プロトコルがハイパーテキスト転送プロトコルであり、連結が、ハイパーテキスト・マークアップ言語ファイルおよび共通ゲートウェイ・インターフェース・プログラムの1つへのハイパーリンクであることを特徴とする、請求項40に記載のコンピュータ・システム。

【請求項48】前記組込みステップが、識別済み連結と引数として渡された状態情報によりCG I変換プログラムを呼び出すためにHTMLファイルを求める要求である識別済み連結を変更するステップをさらに含むことを特徴とする、請求項47に記載のコンピュータ・システム。

【請求項49】前記組込みステップが、識別済み連結と引数として渡された状態情報によるCG Iプログラムへの呼出しである識別済み連結を変更するステップをさらに含み、前記組込みステップがCG Iプログラムによって実行されることを特徴とする、請求項47に記載のコンピュータ・システム。

【請求項50】前記組込みステップが、識別済み連結によりCG I変換プログラムを呼び出すためのCG Iプログラムへの呼出しである識別済み連結と、CG Iプログラムに関連する引数の数を示す引数カウンタと、引数として渡された状態情報とを変更するステップをさらに含み、前記組込みステップが変換プログラムによって実行されることを特徴とする、請求項47に記載のコンピュータ・システム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、コンピュータおよびコンピュータ・ネットワークに関する。詳細には、本発明は、無状態プロトコル(stateless protocol)を介してネットワーク上で通信しながら、状態を保存するコンピュータに関する。さらに具体的には、本発明は、ハイパーテキスト転送プロトコル(HTTP)を使用して、インターネット、特にワールド・ワイド・ウェブ上で通信するコンピュータで状態を保存するための方法およびシステムに関する。

【0002】

【従来の技術】

関連出願

「無状態ネットワーク・プロトコルにおける状態の保存」という名称の本出願は、チュウ(Chiu)他により1995年8月7日に出願され、「A Method for Creating a Hypertext Language for a Distributed Computer Network」という名称の第512205号(IBM整理番号ST995025)と、ラガード(Lagarde)他により1995年6月7日に出願され、「Computer Network for WWWServer Data Access over Internet」という名称の第474571号(IBM整理番号PO995020)という関連米国特許出願に関連する。これらの出願は、本出願人であるIBMに譲渡されたものであり、全体を参考することにより本出願に組み込まれる。

【0003】用語集

本明細書で使用する所与の用語は辞書の意味も暗示するが、用語によっては以下の用語解説が有用である場合もある。

【0004】インターネット

TCP/IPプロトコル・セットを使用するネットワークおよびゲートウェイからなるネットワーク。

【0005】TCP/IP

伝送制御プロトコル/インターネット・プロトコル。電子メールからビデオまで、インターネットが扱うデータを分割し、経路指定し、再構築するためにインターネットが使用するパケット交換方式。

【0006】クライアント

クライアントとは、コマンドに関連するタスクを実行するサーバに対してコマンドを出すコンピュータである。

【0007】サーバ

別のコンピュータの命令でタスクを実行するコンピュータがサーバである。ウェブ・サーバは、通常、1つまたは複数のクライアントをサポートする。

【0008】ワールド・ワイド・ウェブ(WWWまたはウェブ)

強調表示された該当単語または句(ハイパーリンク)上でクリックすることにより、インターネット上で情報を探している人が、サーバ間およびデータベース間を切り

替えることができるようにするインターネットのアプリケーション。インターネットWWWサーバは、複数のクライアントをサポートし、情報を提供する。ウェブは、URLでアドレス指定されたすべての資源を備えたインターネットと見なすことができ、HTMLを使用してURLに対応する情報を表示し、他のURLに対するポイント・クリック・インターフェースを提供する。

【0009】URL(Universal Resource Locator)

インターネット上の情報を明確に識別またはアドレス指定するための方法。電子メール・アドレスのウェブ文書バージョンであると見なすことができる。URLは、他のものに深く埋め込まれた文書に属する場合、厄介なものになる可能性がある。URLへのアクセスは、ハイパーリンクによって行うことができる。URLの一例は「http://www.arun.com:80/table.html」である。URLは4つの構成要素を有する。左から始めて第1の構成要素は、使用するプロトコルを指定するものであり、「:」によってロケータの残りの部分から分離されている。次の構成要素は、ターゲット・ホストのホスト名またはIPアドレスであり、これは左側の「//」と右側の「/」または任意で「:」によって区切られている。ポート番号は任意であり、左側は「:」によってホスト名から区切られ、右側は「/」によって区切られている。

4番目の構成要素は、実際のファイル名またはプログラム名である。この例の「.html」という拡張子は、これがHTMLファイルであることを意味する。

【0010】ハイパーリンク(またはハイパーテキスト・リンク)

単語、句、アイコン、またはピクチャに埋め込まれたネットワーク・アドレスであって、それを選択したときに活動化されるもの。その項目に関する情報は、クライアントに返され、ウェブ・ブラウザを使用して表示される。

【0011】ハイパーテキスト・マークアップ言語(HMTL)

HTMLは、ウェブ・クライアントが表示する文書を作成し接続するためにウェブ・サーバが使用する言語である。HTMLはハイパーテキスト文書を使用する。ハイパーテキスト文書のその他の使い方については、バーンスタイン(Bernstein)他に対して1993年4月20日に付与された米国特許第5204947号、バーンスタイン他に対して1994年3月22日に付与された米国特許第5297249号、ルイス(Lewis)他に対して1994年10月11日に付与された米国特許第5355472号に記載されているが、これらの特許はいずれもIBMに譲渡され、参照により本出願に組み込まれる。

【0012】ハイパーテキスト転送プロトコル(HTTP)

HTTPは無状態プロトコルの一例であり、クライアン

トからサーバへのあらゆる要求が独立して扱われることを意味する。サーバは以前の接続を一切記録していない。URLの先頭に「http:」がある場合は、そのファイルがハイパーリンクを含むことを示している。

【0013】ホーム・ページ

インターネット上で組織などに関する蓄積情報までウェブ・ユーザを案内するためのマルチメディアの目次。

【0014】ウェブ・ブラウザ

コンピュータ上で動作するプログラムであって、インターネット・ツリー・ガイドとして機能するもので、ユーザがインターネットで「ネット・サーフィン」する際に使用する絵画デスクトップ、ディレクトリ、検索ツールによって完成する。本出願のウェブ・ブラウザは、ワールド・ワイド・ウェブと通信するクライアント・サービスである。

【0015】HTTPデーモン (HTTPD)

ハイパーテキスト・マークアップ言語および共通ゲートウェイ・インターフェース機能を有するIBM OS/2のウェブ・サーバまたはその他のサーバ。HTTPDは、通常、TCP/IP結合など、インターネット上のマシンへのハードウェア接続およびインターネットへのアクセスを可能にするアクセス・エージェントによってサポートされる。

【0016】連結 (Continuation)

ハイパーテキスト・リンク（またはハイパーリンク）は、クライアント／サーバ通信における連結の例である。連結とは、クライアントがサーバに送信できる新しい要求である。クライアントがサーバから何かを要求する場合、サーバはその応答に1つまたは複数の連結を含めることができる。サーバは、要求に応答する場合、有効な要求になりうる1つまたは複数の連結を含むことができる。しかし、一般に、有用な連結は、元の要求に対して論理的に関連するものである。

【0017】会話 (conversation)

クライアントとサーバとの間の一連の通信であって、サーバは1組の連結によって各要求に応答し、クライアントは必ずその1組の連結から次の要求を選択する。ウェブ上では、ハイパーテキスト・リンクが連結を表し、クライアントはハイパーテキスト・リンクをたどるときに会話に従事する。

【0018】背景 (background)

ネットワークにより、人々が計算を行う方法が変容した。パソコン・コンピュータまたはワークステーションにアクセス可能な人は、インターネットに接続し、世界中のシステムや人々と通信することができる。ワールド・ワイド・ウェブ (WWWまたはウェブ) は、リンクした文書を介して地球全体に分散している豊富な情報へのアクセス権をユーザに提供するような、インターネットの使い方の1つである。また、WWWにより、ユーザは、遠隔サーバ上で動作するプログラムを実行すること

もできる。この機能により、ユーザは、ハードウェアまたはソフトウェアあるいはその両方の制約のためにユーザがローカルで実行することができないプログラムから結果を得ることができ。また、ワールド・ワイド・ウェブ上に遠隔格納されたプログラムをダウンロードして実行することも可能である。これは、ワールド・ワイド・ウェブに接続されたコンピュータにとって使用可能なソフトウェアの量を大幅に増大させる潜在能力を備えている。

10 【0019】ネットワーク・プロトコル

ネットワーク・プロトコルは、マシン同士が互いに通信するための標準的な方法を提供するものである。プロトコルは、ネットワークにより送受信するためにデータをどのようにフォーマットすべきかを示す。異種マシン同士は、標準プロトコルによりネットワーク上でシームレスに通信することができる。標準的なインターネット・プロトコルの例としては、HTTP (たとえば、T. バーナーズ=リー (Berners-Lee) 、R. フィールディング (Fielding) 、およびH. フリスティック (Frystyk) による「Hypertext Transfer Protocol -- HTTP/1.0」(1995年9月4日) 、<http://www.ics.uci.edu/pub/ietf/http/draft-ietf-http-v10-spec-03.html>を参照) 、SMTP (たとえば、「Simple Mail Transfer Protocol」(RFC 821、J. B. ポステル (Postel) 、Information Sciences Institute、USC、1982年8月) 、<http://ds.internic.net/std/std10.txt>を参照) 、NNTP (たとえば、「Network News Transfer Protocol: A Proposed Standard for the Stream-Based Transmission of News」(RFC 977、B. カンター (Kantor) およびP. ラップスリー (Lapsley) 、UC San Diego and UC Berkeley、1986年2月) 、<http://ds.internic.net/rfc/rfc977.txt>を参照) 、FTP (たとえば、J. ポステルおよびJ. K. レイノルズ (Reynolds) による「FileTransfer Protocol (FTP)」(RFC 959、Information Sciences Institute、USC、1985年10月) 、<http://ds.internic.net/std/std9.txt>を参照) 、Gopher (たとえば、F. アンクレサリア (Anklesaria) 、M. マッカヒル (McCahill) 、P. リンドナー (Lindner) 、D. ジョンソン (Johnson) 、D. トリー (Torrey) 、およびB. アルベルティ (Alberti) による「The Internet Gopher Protocol: A distributed document search and retrieval protocol」(RFC 1436、University of Minnesota、1993年3月) 、<http://ds.internic.net/rfc/rfc1436.txt>を参照) 、WAIS (たとえば、F. デーヴィス (Davis) 、B. カーレ (Kahle) 、H. モリス (Morris) 、J. サレム (Salem) 、T. シエン (Shen) 、R. ワン (Wang) 、J. スイ (Sui) 、およびM. グリンbaum (Grinbaum) による「WAIS Interface Protocol Prototype Functional Specification」(v 1.5) (Thinking Machines Corporation、199

0年4月)を参照)などがある。

【0020】クライアント/サーバ・モデルは、ネットワーク・プログラミングで有力なパラダイムの1つを構成する。たとえば、W. R. スティーヴンス (Stevens)による「Unix Network Programming」(Prentice Hall PTR, Englewood Cliffs, NJ, 1990年)や、D. E. コマー (Comer)による「Internetworking with TCP/IP」(vol 1., Prentice Hall, Englewood Cliffs, NJ, 1991年)などを参照されたい。これらは、その全体が参照により本出願に組み込まれる。サーバ・プログラムは、ネットワーク上で複数のユーザによってアクセス可能なサービスを提供する。プログラムは、サーバにメッセージを送り、サーバからの応答を待つ場合、クライアントになる。クライアント・プロセスは、通常、ユーザ対話用に最適化されているが、要求されたサービスまたはサーバの詳細な作業を把握する必要なしに、要求されたサービスを使用する。ワールド・ワイド・ウェブ上の「ブラウザ」はクライアント・プログラムを構成し、ブラウザに情報を送り返すプログラムはサーバ・プログラムを構成する。

【0021】クライアントとサーバは、同期または非同期のいずれかで通信することができる。同期通信では、クライアントはサーバからの応答を待ってから次の要求を出す。非同期通信では、クライアントは、前の要求からサーバへの1つまたは複数の応答を受け取る前にサーバに要求を出すことができる。

【0022】クライアントとサーバとの間の多くのネットワーク・プロトコルは無状態である。これは、クライアントからサーバへのあらゆる要求が独立して扱われることを意味する。サーバは以前の接続を一切記録していない。HTTPは無状態プロトコルの一例である。無状態プロトコルを使用する場合の2つの利点は効率と単純さである。しかし、クライアントとサーバとの間の通信中に状態情報を管理する方が望ましい状況もある。このようなタイプの対話では、プロトコルが無状態であるために問題が発生する可能性もある。

【0023】HTTPプロトコルとワールド・ワイド・ウェブ

本発明の最も当たり前の応用例は、HTTPプロトコルによりワールド・ワイド・ウェブをブラウズする場合である。たとえば、T. バーナーズ=リー、R. フィールディング、およびH. フリースティクによる「Hypertext Transfer Protocol-- HTTP/1.0」(<http://www.ics.uci.edu/pub/ietf/http/draft-ietf-http-v10-spec-03.html>、1995年9月4日)を参照されたい。これは、その全体が参照により本出願に組み込まれる。しかし、当業者であれば、本発明がHTTPに限定されないことが分かるだろう。ウェブの関連様式や、HTTPなどのプロトコルが無状態であることによる制約については、以下に説明する。

【0024】ワールド・ワイド・ウェブは、まとめてネットワーク化された複数のサーバから構成されている。

クライアントは、通常、Netscape社による「NETSCAPE NAVIGATOR」、NCSAによる「MOSAIC」、IBMによる「WEB EXPLORER」という商標で販売されているような標準的なブラウザを使用してサーバと通信する。クライアントとサーバとの間の最も一般的な通信方法は、HTTPプロトコルによるものである。HTTPでは、クライアントは、ファイルを要求するか、またはサーバ上で実行される共通ゲートウェイ・インターフェース(CGI)

プログラムと呼ばれるプログラムを呼び出すことにより、サーバからデータ入手することができる。CGIプログラミングは当技術分野では周知のものである。たとえば、ジョン・ディセンバー (John December) およびマーク・ギンスバーグ (Mark Ginsburg) による「HTML and CGI Unleashed」(Sams.net Publishing、米国インディアナ州インディアナポリス、1995年)などを参照されたい。サーバは、ファイルまたはCGIプログラムからの出力をクライアントに送信する。サーバは、通常、クライアントがアクセスできるファイルおよびプログラムを制限する。

【0025】サーバは、ハイパーテキスト・マークアップ言語 (HTML) を使用してクライアントに情報を送るが、たとえば、イアン・S. グレアム (Ian S. Graham) による「The HTML Sourcebook」(John Wiley & Sons, Inc.、ニューヨーク、1995年)などを参照されたい。これは、その全体が参照により本出願に組み込まれる。HTML文書は、表示すべき情報の間にHTMLマークアップ・タグが散在する従来のASCIIテキストで構成される。このようなタグは、より大とより小の不等号 (< . . . >) で囲まれ、文書の各種部分をどのように解釈するかをブラウザに指示するものである。ブラウザは、インターネット上の情報を明確に識別するかまたはアドレス指定するためにURLを使用する。ブラウザは、マークアップ・タグに格納された命令に従うことにより、URLに対応するHTML文書を読み取り、それを表示する。

【0026】以下に示すHTMLコード・シーケンス(表1)は、1996年6月3日のIBM T. J. Watson研究所のウェブ・ホーム・ページに対応するHTMLテキストを示している。このウェブ・ページは、「<http://www.watson.ibm.com/>」というURLに対応する。このページにアクセスしている標準のブラウザに表示されると思われる対応出力については、図1に示す。

【0027】<HTML><HEAD>
<TITLE>IBM T. J. Watson Research Center home page</TITLE>
<meta name="owner" content="calyson@watson.ibm.com">
50 <meta name="review" content="19960202">

15

```

</HEAD>
<BODY>
<IMG SRC="/watson/mast.gif" alt="Research">
<p>
<h1>IBM T. J. Watson Research Center</h1>
<p>
<IMG SRC="/watson/night.gif" ><IMG SRC="/watson/ha
w2.gif" >
<br>
<i>T. J. Watson Research Center: Yorktown (left) an
d Hawthorne.</i>
<p>
<ul>
<IMG align=middle SRC="/watson/bullet.gif" ><a HRE
F="/watwel.html" >Welcom!</a>
<br>
<IMG align=middle SRC="/watson/bullet.gif" ><a HRE
F="/leo" >Local Education Outreach</a>
<br>
<IMG align=middle SRC="/watson/bullet.gif" ><a HRE
F="/menu.html" >Visitor info and local site direct
ions</a>
<br>
<IMG align=middle SRC="/watson/bullet.gif" ><a HRE
F="/lodging.html" >Local hotels</a>
<br>
<IMG align=middle SRC="/watson/bullet.gif" ><a href
="http://www.ibm.com">IBM home page</a> -- <a href
="http://www.research.ibm.com">IBM Researchhome p
age</a>
<br>
</ul>
<p>
<hr>
<a href="/watson/mail.html" ><IMG align=middle
SRC="/research/images/mail.gif" ></a><b>Click on i
con to send your comments.</b>
<p>
Or, contact<i>webmaster@watson.ibm.com</i>
<p>
<hr>
<Address><homepage@watson.ibm.com></address>
<b>
<A href="http://www.ibm.com/">IBM home page</a>|
<A href="http://www.ibm.com/Orders/">Order</a>|
<A href="http://www.austin.ibm.com/search/">Search
</a>|
<A href="http://www.ibm.com/Assist/">Contact IBM</
a>|

```

10

表1 : IBM T. J. Watson研究所のホーム・ペ
ージに対応するHTMLソース・コード

```

<A href="http://www.ibm.com/Finding/">Help</a>|
<A href="http://www.ibm.com/copyright.html">(C)</a
>|
<A href="http://www.ibm.com/trademarks.html">(TM)<
/a>
]
</b>
</BODY>
</HTML>

```

20

【0028】多くのウェブ・ブラウザにより、ユーザ
は、表示中の文書のHTMLソース・コードを表示させ
ることができる。表1のHTMLテキストは、IBM
T. J. Watson研究所にあるウェブ・サーバにアクセ
ス可能なファイルに格納されている。このウェブ・サ
ーバは、「<http://www.watson.ibm.com/>」というURLの
要求を受け取ると、適切なファイルをクライアントのブ
ラウザに送る。次にクライアントのブラウザは、そのH
TMLファイルを読み取り、表示する。（表1はいくつ
かの関連リンクを含んでいる。ハイパーテキスト・リ
ンクおよびイメージ・ファイルは、そのファイルが特定の
ディレクトリに格納されている場合のみ有効である。た
とえば、表1の「night.gif」ファイルが任意の位置に
格納されている場合、ハイパーテキスト・リンクは無効
になり、関連イメージは表示されなくなる。）

30

【0029】表1の「Visitor info and local site di
rections」という行は、ハイパーテキスト・リンク（ハ
イパーアクションともいう）の一例である。標準のブラウザ

40

によって表示されると思われる対応出力については図1
に示す。ブラウザによって表示されたときに図1に示す
ようにこのリンクでユーザーがクリックすると、新しいH
TMLファイル「menu.html」がサーバから取り出さ
れ、ブラウザによって表示される。ローカル・サーバ上
と遠隔サーバ上の文書へのハイパーテキスト・リンク
は、どちらも1つのHTMLファイルに入れることができる。
世界中のサーバ上の文書をリンクするためにハイ
パーアクションをHTMLファイル内に組み込む能力は、ワ
ールド・ワイド・ウェブの主な特徴の1つである。すな
わち、ウェブ・ブラウザを使用すると、ハイパーテキス
ト・リンクをポイントし、クリックするだけで、世界中
のサーバからの情報をアクセスすることができる。

50

【0030】ハイパーテキスト・リンクはクライアント
／サーバ通信の「連結」の例であることを思い出して
いただきたい。連結とは、クライアントがサーバに送るこ
とができる新しい要求である。クライアントがサーバか
ら何かを要求する場合、サーバは、その応答に1つまたは
複数の連結を含めることができる。連結は有効な要求
を表すことができるはずである。しかし、一般に、有用
な連結は元の要求に論理的に関連している。良好な1組

の連結により、クライアントがサーバと同期通信することが容易になる。各要求後、サーバは1組の連結で応答する。クライアントは次の要求のためにその連結のうちの1つを選ぶ。「会話」とは、クライアントとサーバとの間の一連の通信であり、そこでサーバは1組の連結で各要求に応答し、クライアントは必ず1組の連結から次の要求を選択する。

【0031】ウェブ上では、ハイパーテキスト・リンクが連結を表し、クライアントはハイパーテキスト・リンクをたどるときに会話を従事する。クライアントがハイパーテキスト・リンクをたどらずに、新しいURLを明示的に要求することによって新しいページを獲得すると、会話は中断される。中断した会話に対応するページが依然としてクライアントにとって使用可能である場合、たとえば、ブラウザ・キャッシュ内またはディスク・メモリ内にある場合、中断した会話を続行することは可能である。そうである場合、そのページを再ロードし、引き続きハイパーテキストをたどることにより、会話を続行することができる。クライアントは、同一会話中に複数のサーバと通信することができる。

【0032】より形式的には、以下のような場合、一連のHTMLページであるp₁、p₂、…、p_nによって会話が構成される。

1. p₁、p₂、…、p_nがすべてクライアントによって表示された場合。
2. 1 < i ≤ n になるようなすべての i について、ページ p_{i-1}上のハイパーテキスト・リンクをたどることによって、ページ p_iが獲得された場合。

【0033】非中断会話では、クライアントは、「後戻り」せずにページ p₁から p_nまで獲得するために n-1 のハイパーテキスト・リンクをたどるだけである。中断会話では、クライアントは少なくとも1回は後戻りする。後戻りとは、クライアントが次のように行うことである。

1. まず、ページ p_iを訪れる。ただし、1 ≤ i < n である。
2. ハイパーテキストをたどるかまたは明示的にURLをアドレス指定することにより、他のページを表示する。
3. メモリから p_iを再ロードすることにより、ページ p_iに戻る (p_iがまだ使用可能であると仮定する)。

【0034】すべてのURL要求は無状態である。クライアントが1つのページを複数回要求しても、サーバは前の接続の履歴または知識を管理していない。クライアントがHTMLファイルを要求する場合、クライアントがその要求によって追加の情報を通信するための手段はまったくない。したがって、ウェブ環境では、クライアントがHTMLファイルをブラウズしている間に会話を全体にわたって状態情報を保存する必要性が存在する。本発明はこのような必要性に対処するものである。

【0035】たとえば、ビジネス・トランザクションを

処理するサーバについて検討する。正常に機能するためには、サーバは、クライアントのユーザーIDや、現行トランザクション番号に対応するトランザクション番号などの状態情報を必要とする。したがって、会話においてハイパーテキストをたどることにより、クライアントがHTMLファイルをブラウズする間、この情報を保存する必要性がある。本発明はこのような必要性に対処するものである。

【0036】ウェブ上で状態を処理するための現行方法

- 10 10 ウェブ上で状態を処理するための現行方法の1つは、CGIプログラムの使用を必要とする。クライアントは、それに引数を渡すことによってCGIプログラムを呼び出すことができる。たとえば、<http://tranman.watson.ibm.com/cgi-bin/get-args?var1=7&var2=10>というコマンドは、変数 var1 = 7 および var2 = 10 を渡してCGIプログラムを呼び出すものである。クライアントが変数をCGIプログラムに渡すために正確な構文に従うと予想することは厄介なことである。もっと使いやすい方法は、HTMLの「用紙」によりユーザが引数を入力できるようにする方法である。ウェブ・ブラウザによって表示されるHTML用紙の例は図2に示す。ユーザは、適切なフィールドに記入し、送信ボタンをクリックすることによってサーバに情報を送信する。ユーザがタップした値は、引数としてCGIスクリプトに渡される。「用紙」は、CGIプログラムに引数を渡すための便利なインターフェースになる。クライアントは、呼び出しているCGIプログラムの詳細や、プログラムが予想する引数のフォーマットを知っている必要がない。
- 20 20 【0037】用紙により、クライアントはサーバに状態変数を渡すことができる。また、サーバも用紙を使用してクライアントに変数を渡すことができる。用紙は、クライアントに対して表示されず、クライアントが用紙を受渡ししたときにサーバに戻される隠し変数を含むことができる。通常、ウェブ・サーバは、用紙内の隠し変数として状態変数を渡すことにより、状態を保存する。クライアントが用紙を受渡しすると、用紙を受け取るサーバは、隠しフィールドから状態変数を得ることができる。
- 30 30 【0038】たとえば、ビジネス・トランザクション・サーバがクライアントと通信していると想定する。トランザクション・サーバは、クライアントとの会話の残りの部分のためにクライアント・ユーザーIDとセッションIDを入手する必要がある。サーバは、クライアントから受渡しされた用紙からクライアントのユーザーIDを得ることができる。この用紙はCGIプログラムを呼び出し、次にそのプログラムがセッションIDを生成する。サーバからの後続応答はそれぞれ1つの用紙になる。この用紙は、動的に生成されるものであり、隠し変数として組み込まれたユーザーIDとセッションIDを含む。クライアントは、サーバが生成した用紙を完成し、受渡し
- 40 40
- 50 50

することによって応答する。

【0039】図3は、HTML用紙を使用して状態を保存するための現行方法の例を示している。図示の通り、サーバ410は、動的に生成されたHTML用紙420への隠し引数に状態変数を組み込む。状態変数425は、クライアント450とサーバ410との間でやりとりされる。クライアント450とサーバ410は、用紙を使用して状態情報425をやりとりする。サーバ410は、大急ぎでHTML用紙420を作成し、隠しフィールドに状態変数425を組み込むことにより、クライアントに状態情報を渡す。クライアント450は、サーバ410が生成した用紙420'を完成し、受渡しすることにより、サーバに状態情報425を返す。

【0040】状態を処理するための現行技術の限界
上記の手法の問題は、会話中のクライアントとサーバとの間の対話のタイプが厳重に制限されることである。サーバ410は、必ず、隠し変数425を含む動的に生成したHTML用紙420によってクライアント450に応答しなければならない。クライアントがHTMLファイルをブラウズしている間、状態を保存するための手段はまったくない。たとえば、クライアントがセッションの途中でカタログをブラウズしたいと望んでいると想定する。このカタログはHTMLファイルから構成されている。現行技術を使用すると、クライアントが状態情報を失わずにカタログ（内の各種HTMLファイル）をブラウズできるようにするための手段はまったくない。サーバにより、クライアントがカタログを表示させることによって会話を続行できるようになっている場合、クライアントがカタログからHTMLファイルにアクセスすると、ただちに状態情報が失われる。

【0041】したがって、状態情報を保存しながら、クライアントがカタログをブラウズできる、すなわち、各種HTMLファイルにアクセスできるようにするためのシステムおよび方法の必要性が存在する。本発明は、カタログを構成するHTMLファイルが各種サーバ上に存在しているかどうかにかかわらず、このような必要性に対処するものである。

【0042】状態を保存するための現行技術の限界については、これ以外にも注目されている。たとえば、「Persistent Client State HTTP Cookies」（Netscape Communications Corporation、1996年、http://home.netscape.com/newsref/std/cookie_spec.html）を参照されたい。また、「Proposed HTTP State-Info Mechanism」（D. M. クリストル（Kristol）、AT&T Bell Laboratories、1995年9月22日、<http://www.research.att.com/~dmk/session01.txt.>）、M. カトラー（Cutler）およびD. ホール（Hall）による「August 1995 Web Watch」（<http://www.netgen.com/corpinfo/press/webwatchv6n8.html>）も参照されたい。状態を保存するためにHTTPプロトコルを変更しそうなKristol提案の解決

策とは異なり、本発明では、基礎となるプロトコルに変更を加える必要なしに、状態を保存する。

【0043】Netscape Communications社によるもう1つの解決策は、同社のブラウザにクッキー（Cookies）と呼ばれる機能を追加することであった。「Persistent Client State HTTP Cookies」（Netscape Communications Corporation、1996年、http://home.netscape.com/newsref/std/cookie_spec.html）を参照されたい。この場合、サーバは、クッキーと呼ばれる状態オブジェクトをその応答に付加することにより、HTTP要求を満足することができる。クッキーは、その状態が有効になるURLの範囲の記述を含む。また、クッキーは、クライアント上で動作しているNetscapeブラウザによって格納される。今後、クッキーに指定されているURLの1つにクライアントがHTTP要求を行うと、その要求には、クッキーに格納された状態オブジェクトをクライアントからサーバに返送することが含まれることになる。

【0044】この手法には、いくつかの欠点がある。状態の保存を望んでいるサーバ・アプリケーションは、その状態を使用する可能性のあるすべてのURLのリストを提供しなければならない。これは厄介なことであり、ときには不可能な場合もある。また、クッキーには、状態情報と特定の会話を相関させる方法が欠落している。たとえば、ブラウザが2つの別々の会話で同一URLにアクセスしていると想定する。第1の会話では、状態情報は、URLにアクセスしたときに存在し、クッキーを介してサーバに渡される。第2の会話では、URLにアクセスしたときに状態情報は一切存在しない。しかし、前のクッキーが依然として存在し、前の状態が依然としてサーバに渡される。これは、サーバを混乱させ、前の状態情報が依然として新しい会話に適用されると思わせる恐れがある。もう1つの問題は、クッキーが標準の機能ではなく、Netscapeのプロトコルをサポートするサーバおよびブラウザでしか機能しないことである。

【0045】

【発明が解決しようとする課題】したがって、状態情報の使用を必要とするURLのリストに限定されず、古い状態情報をサーバに渡すという問題を回避するために状態情報が特定の会話と相関されるような、無状態プロトコルで状態を保存するための方法およびシステムの必要性が存在する。しかも、HTTPプロトコルをサポートするブラウザで機能し、クライアントまたはサーバ上で標準外の特殊機能を必要としないような、HTTPなどのプロトコルで状態を保存するシステムの必要性が存在する。

【0046】

【課題を解決するための手段】本発明は、上記の必要性に応じて、無状態プロトコルを使用してネットワーク上で通信するコンピュータの状態を保存するための方法お

よりシステムに関する。好ましい実施例はハイパーテキスト転送プロトコル（H T T P）を使用してワールド・ワイド・ウェブ（W W W またはウェブ）により通信するコンピュータ用であるが、本発明は、他の形式のネットワーク化通信にも適用される。

【0047】クライアントのためにサーバが実行するサービスは、クライアントが呼び出すプログラムであると想定する。1つのサービスは可変数の引数を受け入れることができる。会話は、サービスに関するクライアントと1つまたは複数のサーバとの間の一連の通信であり、サーバからの各応答は別のサービス要求を可能にする1つまたは複数の連結を含み、クライアントは会話を続行するためにその連結のうちの1つを呼び出さなければならない。

【0048】したがって、無状態プロトコルによりクライアントに対してネットワーク化された1つまたは複数のサーバからサービスを要求できるようになっているクライアントとの間の会話で状態情報を保存するような本発明の機能を有するコンピュータ化した方法、システム、コンピュータ・プログラム製品は、クライアントが無状態プロトコルを使用してサーバとの会話を開始することと、サービス要求が状態情報の保存を必要とする時期を検出することと、状態を保存すべきときに、サービスを実行し、サービスからの出力に含まれるすべての連結を識別することと、すべての識別済み連結に再帰的に状態情報を組み込むことと、出力をクライアントに通信することとを含み、会話の持続期間中に状態情報が保存され、すべてのサービスに提供される。

【0049】本発明の他の態様によれば、状態情報の組込みは、サーバによって実行され、サーバからクライアントに通信される。本発明の他の態様は、サーバに結合されたメモリに状態情報の少なくとも一部を格納することと、すべての識別済み連結に状態情報の一部を表すインデックスを組み込むこととを含む。

【0050】本発明のさらに他の態様は、やはりクライアントに通信されるサービスからの出力に状態情報を組み込むために、コンピュータ・プログラム・コードをクライアントに動的にダウンロードすることを含む。本発明のさらに他の態様は、クライアントに結合されたメモリに状態情報の少なくとも一部を格納することと、格納済みの状態情報を表すインデックスを組み込むこととを含む。

【0051】好ましい実施例の方法により、ワールド・ワイド・ウェブ上でウェブ・ブラウザを使用してハイパーテキスト・リンクを移動する間に状態を保存することができる。ハイパーテキスト・リンクは連結を構成する。クライアント・ブラウザは、新しいページを取り出すためにハイパーテキスト・リンクをたどることにより、会話を追跡する。本発明は、会話により状態変数を保存するような機能を有する。本発明の一態様によれ

ば、会話全体にわたって保存すべき状態変数は、会話全体にわたって呼出し可能なあらゆるC G I プログラムに渡される。

【0052】ワールド・ワイド・ウェブを介してクライアントとサーバをネットワーク化し、無状態プロトコルがハイパーテキスト転送プロトコル（H T T P）であり、連結が、ハイパーテキスト・マークアップ言語（H T M L）ファイルおよび共通ゲートウェイ・インターフェース（C G I）プログラムの1つへのハイパーリンクで10ある場合、本発明は、所定の基準に応じてサービスからのデータ出力とハイパーリンクのフィルタ処理または追加あるいはその両方を可能にするような機能を有する。状態情報を組み込むための本発明のさらに他の態様は、識別済み連結と引数として渡された状態情報によりC G I 変換プログラムを呼び出すためにH T M Lファイルを求める要求である識別済み連結を変更することを含む。状態情報を組み込むための本発明のさらに他の態様は、識別済み連結と引数として渡された状態情報によるC G I プログラムへの呼出しである識別済み連結を変更することを含み、組み込みステップはC G I プログラムによって実行される。状態情報を組み込むための本発明の他の態様は、識別済み連結によりC G I 変換プログラムを呼び出すためのC G I プログラムへの呼出しである識別済み連結と、C G I プログラムに関連する引数の数を示す引数カウンタと、引数として渡された状態情報を変更することを含み、組み込みステップは変換プログラムによって実行される。

【0053】本発明の上記およびその他の特徴および利点は、添付図面に関連して以下に示す詳細な説明から明らかになるだろう。

【0054】

【発明の実施の形態】本発明は、無状態プロトコルによりネットワーク上で通信するコンピュータ用の機能付与技術である。好ましい実施例はハイパーテキスト転送プロトコル（H T T P）を使用してワールド・ワイド・ウェブ（W W W またはウェブ）により通信するコンピュータ用であるが、本発明は、他の形式のネットワーク化通信にも適用される。

【0055】図4は、無状態プロトコルを使用して状態40を保存するための本発明による一般的な方法を示すが、これはワールド・ワイド・ウェブ上で通信するクライアントとサーバに制限されるものではない。後述する好ましい実施例は、特にワールド・ワイド・ウェブに適用される。この場合、簡略化のため、クライアントのためにサーバが実行するサービスは、クライアントが呼び出すプログラムであり、1つのサービスは可変数の引数を受け入れができるものと想定する。

【0056】図示の通り、ステップ500では、クライアントがサーバからサービスを要求する。ステップ5050、510、515によって表される経路は、提供され

23

るサービスが状態保存を必要としない場合に取られるはずである。ステップ510では、任意の時点で、サーバは要求を処理し、その要求に関して、現行会話で呼出し可能なすべてのサービスにとって状態変数が使用可能である必要があるかどうかをサーバが判定する。次にサーバは、その出力と、<state-variable-list>で示される状態変数のすべてを、convert1で示されるプログラムに*

```
servicel <service-arg-list>
```

ただし、servicel 605はサービスであり、<service-arg-list>はクライアントが連結を選んだ場合にサービスに渡されるサービス引数610のリストである。図6※

```
convert2 service-string <state-variable-list>,
```

ただし、convert2 650は特殊サービス（後述する）への呼出しであり、service-string 680はservicel 605および<service-arg-list> 610と、図6に示すconvert2への呼出しにおいてサービス引数610を<state-variable-list> 670から区別するための何らかの区切り情報685とを含むストリングである。<state-variable-list> 670は、会話の持続期間中に保存し、すべてのサービスに使用可能なものにするべき状態情報を表す。

【0058】もう一度図4を参照すると、ステップ525でクライアントは、サーバから送信された出力と変更済み連結とを受け取る。この場合、会話に関する各変更済み連結は、(2)に示すようなconvert2プログラムへの呼出しになる。ステップ540でクライアントは出力を検査する。連結を選択した場合、プロセスはステップ500に戻り、そこで（変更済み）サービス要求がサーバに送られる。ステップ505でサーバは、変更済みサービス要求を処理し、convert2プログラムを呼び出し、処理はステップ530で続行する（前のステップ520の明示的な組込み呼出しによる）。ステップ530で

(図5および図6を参照) convert2プログラムは、サービス605と、service-string 680からサーバに渡された引数(<service-arg-list>) 610とを解析する。convert2プログラム650は、<service-arg-list> 610ならびに<state-variable-list>に関するすべての変数をそれに渡すことにより、要求したサービス605(ここではservicel)を呼び出す。このようにして、servicelは、必要に応じてすべての状態変数にアクセスすることができる。ステップ535でconvert2プログラムは、サービス(servicel)出力を受け取り、その出力と<state-variable-list>をconvert1に渡す。ステップ520でconvert1は、前述のようにそれぞれの連結を変更する。この場合も出力はステップ525でクライアントに通信され、プロセスは、会話の持続期間中に保存された状態情報670を使って反復する。

【0059】好ましい実施例

図7は、ワールド・ワイド・ウェブ上でビジネス・トランザクションを行うための本発明の機能を有するシステ

24

*渡し、処理はステップ520に移行する。ステップ520でconvert1プログラムは、サービスが生成した連結を変更するが、その他のデータはすべて未変更のままクライアントに返す。

【0057】たとえば、図5に示すように、通常の状況下では、プログラムservicelへの呼出しを表す連結は以下のようになるはずである。

(1)

※に示すように、本発明のconvert1プログラムは、各連結
10 (1)を以下の形式になるように変更することにより、
状態を保存する。

(2)

ムの一実施例を示している。ウェブ・サーバ410'は、インターネットにより商品を販売するビジネスを可能にするものである。顧客は、標準のブラウザ460を実行しているクライアント450を介してサーバ410'にアクセスする。安全に通信するため、「ブラウザ」460は、SSLを使用して通信できなければならない。たとえば、A.O. フライヤー(Freier)、P. カールトン(Karlton)、P.C. コヒー(Kocher)による「The SSLProtocol Version 3.0」(Internet Draft, 1996年3月、<http://home.netscape.com/eng/ss13/ssl-toc.html>)を参照されたい。これは、その全体が参考により本出願に組み込まれる。ただし、一部のサービスは、SSLをサポートしないブラウザによって使用することができる。ユーザは、直接アクセス記憶装置(DASD)470などの安定した記憶媒体上に格納可能なカタログをブラウズすることができる。従来のカタログと同様、ユーザは製品説明をブラウズし、自分の購入リストに追加すべき品目を探し、選択することができる。購入リストが完成したと判定すると、ユーザは購入をコミットし、後で請求を受ける。

【0060】図示の通り、サーバ410'は、顧客、在庫、データベース475に格納された製品に関する情報を管理するために、従来のデータベース管理システム(DBMS)412を含むことができる。DBMSの例としては、「DB2」という商標でIBMから販売されているものがある。さらに、サーバ410'は、ユーザが会話の最中に製品カタログをブラウズできるようにする。サーバ410'は、製品カタログのフォーマットについてほとんど想定していない。カタログは、HTMLファイル425ならびに従来のCGIプログラムから構成することができる。このようなファイルやプログラムは、ローカルまたは遠隔サーバに関連付けることができる。ユーザIDおよびセッションIDなどの状態情報は、会話中にサーバ410'とクライアント450との間で管理しなければならない。本発明は、会話中にこの状態情報を透過的に管理するための改良された方法およびシステムを提供する。

50 【0061】クライアント450は、サーバ410'に

25

関連する「ホーム・ページ」にアクセスすると同時に製品カタログを表示させることができる。製品を購入するか、顧客情報を更新するか、または所持のタイプの情報にアクセスするためには、ユーザIDとパスワードを入力することにより、ユーザが認証を行えるようにすることが必要である。本発明によれば、認証は1回の会話当たり1回だけ必要になる。ユーザが認証されると、ただちに本発明の変換プログラム416により、ユーザIDが会話に組み込まれる（保存される）。

【0062】図8は、クライアント450が状態を保存しながらHTTPを使用してサーバ410' と対話するための本発明による方法の一例を示している。図示の通り、ステップ700でクライアントは、サーバ410' 上に存在するホーム・ページにアクセスする。ステップ710でクライアントは、製品カタログのブラウズを開始し、ステップ720で、オンライン製品カタログからハイパーリンクを選択することなどにより、カタログ商品のブラウズを続行する。カタログをブラウズするためだけであれば、認証は一切不要なので、通信は無状態になり、カタログをブラウズできる人数は最大になる。ステップ730では、購入リストに追加すべき品目を見つける。次にステップ740でクライアントは、続行するためにユーザIDとパスワードを入力しなければならない。クライアントがそのシステムに初めてアクセスした場合、クライアントはユーザIDとパスワードを選び、何らかの追加の任意情報（住所、電話番号など）をサーバに提供する。ステップ745で変換プログラム416は、本発明によりユーザIDとセッションIDを会話に組み込む。ステップ750でユーザは、追加の製品を表示させ、追加品目を購入リストに追加し、購入をコミットし、データベース情報を表示し更新することができる。有利なことに、状態変数は保存され、再認証は不要である。状態情報、すなわち、ユーザIDとセッションIDは保存され、会話の残りの部分で呼び出されるあらゆるCGIプログラムにとって使用可能なものになる。

【0063】現行の「用紙」技術を使用すると、ユーザは、購入リストへの新しい品目の追加など、認証を必要とするアクションを試みるたびに、ユーザIDとパスワードを再入力しなければならないことを思い出していたいだきたい。セッションIDの場合は、認証が必要なときにセッションIDを思い出してそれを入力するようサーバがクライアントに指示しなければならぬので、困難がさらに増大するはずである。

【0064】また、「クッキー」を使用すると状態が保存されるURLの範囲が制限されることも思い出していたいだきたい。さらに、クッキーの使用では、古い状態情報がサーバに提供される恐れがある特定の会話と状態情報を相関させる能力が欠落している。最後に、クッキーは、特定のブラウザの使用を必要とし、クライアントまたはサーバ上で特殊機能または標準外の機能を必要と

26

する可能性がある。

【0065】図9は、図7に示すシステムおよび図8の方法ステップ745で状態を保存するための本発明による方法のより詳細な例を示している。サーバ410' は、TCP/IP結合などの典型的なインターネット接続部およびアクセスを含む、従来のウェブ・サーバであり、HTMLおよび共通ゲートウェイ・インターフェース(CGI)機能413をさらに有するものと想定する。

【0066】図示の通り、ステップ740' では、ウェブ・ブラウザ460を実行しているクライアント450が（無状態プロトコル）HTTPを介してサービスを要求するためにウェブ・サーバ410'へのハイパーテリンクを選択したと想定する。ステップ810でサーバ410'は、たとえば、CGIプログラム「p1」415への呼出しとしてそのURLを解釈し、そのプログラムは、会話から呼出し可能なすべてのCGIプログラムにアクセス権が与えられるように「x1、x2、...、xn」などの状態変数を会話に組み込む必要があるかどうかを判定する。p1は、クライアント450が会話を続行するためにハイパーテキスト・リンクを備えたHTMLページ「h」を生成する。出力であるページ「h」を未変更のままクライアントに返す代わりに、「p1」は、変換プログラムのconvert1モジュールに引数「h、x1、x2、...、xn」を渡すことにより、本発明の変換プログラム416を呼び出せるようになっている。convert1への呼出しは、以下のような形式になっているはずである。

convert1 'h, x1, x2, ..., xn'

【0067】ステップ811および812で変換プログラムのconvert1モジュールは、状態変数を保存するためにh内のHTMLへのハイパーテキスト・リンクをすべて変更する。すべての相対ハイパーテキスト・リンクは、絶対ハイパーテキスト・リンク（ハイパーテリンクともいう）に変換される。相対／絶対アドレス変換方式の例については、1995年8月7日に出願され、チュウ他による「A Method for Creating a Hypertext Language for a Distributed Computer Network」という名称で、参照により組み込まれる上記の関連米国特許出願第512205号（IBM整理番号ST995025）を参照されたい。上記のように、これらの出願は本出願人に譲渡されたものである。当業者であれば、HTMLファイルへのリンクの変更（ステップ811）とCGIプログラムへのリンクの変更（ステップ812）が本発明の範囲および精神の範囲内の1パスまたは2パス・プロセスで実行可能であることが分かるだろう。

【0068】図示の通り、ステップ811で変換プログラム416のconvert1モジュールは、HTMLページhを取り、状態変数を保存するためにHTMLファイルへのすべてのハイパーテキスト・リンクを変更する。HTMLファイルへのハイパーテキスト・リンクは、h、x

1、・・・、 $x = n$ からなる引数を指定した CGI プログラム convert2への呼出しになるように変更することができる。図 10 に関連し、たとえば、 h が HTML ファイル 915 「mail.html」への以下の参照を含むことを考慮し、状態変数 930 が $x = 32$ と $y = 45$ であると想定する。

```
<A HREF="http://www.watson.ibm.com/mail.html">
これは、convert1論理によって図 11 に示す形式に変更されるはずである。
```

```
<A HREF="http://www.watson.ibm.com/cgi-bin/convert
2?url=/www.watson.ibm.com/mail.html&x=32&y=45">
```

【0069】ステップ 812 で、変換プログラムの convert1 モジュールは、CGI プログラムへのすべてのハイパーテキスト・リンクを変更する。ただし、CGI プログラムへの呼出しであるハイパーテキスト・リンクは、以下の 2通りの方法で状態変数を保存することができることに留意されたい。

(a) 図 13 に示すように、CGI プログラム 940 に状態変数 930' を渡すが、CGI プログラムによって生成されたハイパーテキスト・リンク内に状態変数を組み込まない。すなわち、convert2への呼出しを組み込まない。CGI プログラムは、この手法を使用して、それが生成したハイパーテキスト・リンク内での状態の伝播を担当する。

(b) 好ましくは、図 15 に示すように、状態変数 930' を CGI プログラム 950 に渡し、CGI プログラムによって生成されたハイパーテキスト・リンク内に
(変換プログラム 920' への組込み呼出しにより) 状態変数を組み込む。

【0070】手法 (a) と (b) の両方を利用するためには、変換プログラムは、本発明の範囲内の様々な技術の 1 つに基づいて、CGI プログラムを区別する方法を決定することができる。たとえば、図 12 に示すように、その名前が「type」 というサブストリングで始まる CGI プログラムはタイプ I の CGI プログラムと見なすことができ、第 1 の方法 (a) を使用して処理されるという、命名規則を使用することもできる。その名前が「type」 というサブストリングで始まらないサブストリングは、タイプ II の CGI プログラムと見なすことができ、第 2 の方法 (b) を使用して処理される。

【0071】たとえば、(図 12 に関連して) タイプ I の CGI 呼出しの以下の例について検討する。

```
<A HREF="http://www.watson.ibm.com/cgi-bin/type1?
arg1=55">
```

ここで、状態変数が $x = 32$ と $y = 45$ であると想定する。変換プログラム 416 は、(図 13 に示すように) 以下の形式になるようにハイパーテキスト・リンクに状態変数 930' を付加するはずである。

```
<A HREF="http://www.watson.ibm.com/cgi-bin/type1?&
arg1=55&x=32&y=45">
```

【0072】次に、図 14 に関連して、タイプ 2 の CGI プログラムへのハイパーテキスト・リンクの例について検討する。

```
<A HREF="http://www.watson.ibm.com/cgi-bin/prog?ar
g1=55">
```

この場合も、状態変数 930' が $x = 32$ と $y = 45$ であると想定する。変換プログラムは、(図 15 に示すように) 以下の形式になるようにこのハイパーテキスト・リンクを変更するはずである。

10 <A HREF="http://www.watson.ibm.com/cgi-bin/convert
2?url=/www.watson.ibm.com/cgi-bin/prog&numargs=1&
arg1=55&x=32&y=45">

ただし、「numargs=1」という引数 970 は、CGI プログラム 950 には最初に 1 つの引数だけが渡され、残りの引数は変換プログラムによって渡される状態変数 930' であることを convert2 に示すものである。次に、変更された出力は要求側クライアントに返される。ステップ 813 でクライアント 450 は、サーバ 410' から HTML ファイル h を受け取る。次に、クライアント 20 戻るあらゆるハイパーテキスト・リンク (タイプ 1 の CGI プログラムの結果であるハイパーテキスト・リンクは除く) は、変換プログラムの convert2 ルーチンへの呼出しになる。ステップ 814 では、ブラウザ 460 を実行しているクライアント 450 がハイパーテキスト・リンクのうちの 1 つを選択する。ステップ 815 でサーバは、選択したハイパーテキスト・リンクが convert2 への呼出しであるかどうかを判定する。YES である場合、プロセスはステップ 817 で続行する。ステップ 817 には、2通りの可能性がある。

30 (1) convert2 に渡される URL は HTML ファイルを参照するものである。この場合、プロセスはステップ 819 で続行する。たとえば、クライアントが以下のリンクを選択したと想定する。

```
<A HREF="http://www.watson.ibm.com/cgi-bin/convert
2?url=/www.watson.ibm.com/mail.html&x=32&y=45">
```

ステップ 819 で convert2 は、ファイル 「mail.html」 に含まれる HTML ページを取り出す。次にこれは、変換プログラムの convert1 モジュールにその HTML ページと状態引数 $x = 32$ 、 $y = 45$ を渡し、プロセスは前述のようにステップ 811 に戻る。

(2) ハイパーテキスト・リンクは CGI プログラムへの呼出しになる。この場合、プロセスはステップ 818 に移行する。たとえば、クライアントが以下のリンクを選択したと想定する。

```
<A HREF="http://www.watson.ibm.com/cgi-bin/convert
2?url=/www.watson.ibm.com/cgi-bin/prog&numargs=1&
arg1=55&x=32&y=45">
```

【0073】この場合、convert2 への第 2 の引数である numargs=1 は、初期ハイパーテキスト・リンクが 「prog」 に 1 つの引数、すなわち、「arg1=55」 しか渡してい

ないことを示している。残りの2つの引数である「x=32」と「y=45」は、変換プログラム416によって組み込まれた状態変数である。convert2は、状態変数を含む、3つの引数をすべてprogに渡す。次にプロセスは、前述のようにステップ811に戻る。

【0074】本発明のこの方法は、クライアント450とサーバ410との間でやりとりされるすべてのハイパーアリンクに状態を組み込むことにより、状態情報を有利に保存する。当業者であれば、サーバ410'に結合されたファイル・システムまたはデータベース425に状態情報のほとんどを格納することにより、クライアントとサーバとの間で通信される詳細なレベルを低減できることが分かるだろう。この場合、必要なことは、クライアントとサーバとの間で状態変数へのインデックス（またはポインタ）をやりとりすることだけである。

【0075】本発明は、後でクライアント上で実行可能なプログラムをサーバからダウンロードすることを必ずしもサポートしていない標準のブラウザ460用に機能するように設計されている。Java（「アプレット」）またはその他のこのような言語を使用して作成されたような、ダウンロード可能なサーバ・プログラムをサポートするブラウザの場合は、追加の機能が可能である。Javaプログラミング環境は当技術分野では周知のものである。たとえば、P.ティマ（Tyma）、G.トロク（Torok）、T.ダウニング（Downing）による「Java Primer Plus」（Waite Group Press、1996年）を参照されたい。これは、その全体が参照により本出願に組み込まれる。また、パトリック・ノートン（Patrick Naughton）による「The Java Handbook」（Osborne McGraw-Hill、1996年）も参照されたい。これも、その全体が参照により本出願に組み込まれる。たとえば、サーバ410'は、クライアントに状態を格納させるような、ダウンロード可能なプログラムを含むことができるはずである。この手法を使用すると、状態の一部または前部がクライアント上に格納できるはずである。上記のように、クライアントから状態を取り出せるようるために、メモリ内の状態情報の位置を参照するインデックスをサーバとクライアントとの間でやりとりすることができます。

【0076】ダウンロード可能なサーバ・コードを本発明に適用するもう1つの例は、「変換プログラム」416がクライアント上で動作できるようにすることである。この場合、クライアントは、実行のためにサーバ410'からそのクライアントに「変換プログラム」論理416の一部または前部をダウンロードするはずである。これにより、本発明の機能が完全に可能になり、処理のすべて（または一部）がクライアント450上でローカルに行われるはずである。もはやクライアントは、会話中にHTMLページをフィルタ処理するために遠隔サーバを調べる必要がなくなり、フィルタ処理はすべて

ローカルに行われる。この場合、サーバにかかる負荷が低減されるという利点がある。さらに、そこからクライアントがアプレット入手するサーバがネットワーク障害のために停止するかまたは使用不能になった場合でも、クライアントは会話を続行することができる。

【0077】その他の実施例

複数の通信サーバ上での状態の保存

当業者であれば、本発明の範囲内で、複数のサーバ上で状態伝播のために複数の変換プログラムを使用できる

10 ことが分かるだろう。たとえば、ウェブによる航空座席予約システムは、状態を管理するために変換プログラム（変換プログラムA）を備えている可能性がある。ハイパーテキスト・リンクの1つは、専用の変換プログラム（変換プログラムH）を備えた遠隔サーバ上のホテル予約システムにつながっている可能性がある。クライアントは航空座席予約システムを使い始めるはずである。任意の時点で、状態情報が会話を付加される。次にクライアントは、ホテル予約システムへのハイパーテキスト・リンクをたどる。変換プログラムAは、クライアントがホテル予約システムを使用している間、状態情報を管理し続ける。すべての状態変数は、ホテル予約システムのCGIプログラムに伝播される。このような遠隔サーバのCGIプログラムは、これらの状態変数を無視するだけである可能性もある。これに対して、ホテル予約システムが航空座席予約システムからの状態変数を理解できる場合、これらの変数はホテル予約システム（変換プログラムH）によって使用できるはずである。

【0078】ホテル予約システムは、任意の時点でその変換プログラム（変換プログラムH）を呼び出して、追加の状態変数を組み込むことができる。これが行われると、変換プログラムHへの呼出しは、変換プログラムAへの呼出し内にネストすることができる。これによって問題が発生することはない。CGIプログラムには、変換プログラムAと変換プログラムHの両方からの引数が渡される。変換プログラムAが変換プログラムHへの呼出しを表すCGI機能を認識する能力を備えている場合、以下のように追加の事柄が可能である。

40 (1) 前述のように、変換プログラムAは変換プログラムHをタイプIのCGIプログラムとして扱うことができる。この場合、変換プログラムAは会話に含まれる今後のハイパーテキスト・リンクの監視を停止することができる。

(2) 変換プログラムAは変換プログラムHをタイプIIのCGIプログラムとして扱い、ハイパーテキスト・リンクを変更するために続行することができる。さらに、変換プログラムAは、ユーザがいずれかの変換プログラムの制御からエスケープできるようにするよう、今後のHTMLページへの特殊リンクを追加することができる。

本発明は、クライアントが会話中にHTMLファイルをブラウズしている間に、クライアントによって表示されたすべてのHTMLテキストをフィルタ処理するためのシステムおよび方法を提供するような特徴も備えている。たとえば、クライアントがサーバに接触し、会話を開始したと想定する。サーバは、すべてのHTMLテキストをフィルタ処理し、反対すべきと判定された句およびハイパーテキスト・リンクを除外することを望んでいる。本発明は、フィルタ処理を実行するサーバにとって遠隔である可能性があるファイルおよびプログラムに対してクライアントがアクセスしている間にHTMLテキストをフィルタ処理または変更するための方法を提供する。

【0080】本発明は、会話中にHTMLページの変更が必要になる様々な応用例に適用することができる。たとえば、サーバ・アプリケーションが、会話中にクライアントに渡されるすべてのHTMLページのフィルタ処理を望んでいると想定する。変換プログラムは、それをクライアントに送る前にHTMLページの不要な部分を変更または除去できるはずである。変換プログラムは、各種サブストリングを求めてテキストを探索するように変更するだけでよいはずである。ただし、変換プログラムは、遠隔サーバ上に存在するCGIプログラムからの出力とページを検閲することができることに留意されたい。クライアントがJavaなどの言語で作成されたプログラムをサーバからダウンロードできる場合、検閲を実行する変換プログラムはクライアント上で実行することができる。

【0081】もう1つの例としては、クライアント450が会話中であり、主要企業の名前がテキスト内に頻繁に現れるものと想定する。変換プログラム416を実行しているサーバ410'は、主要企業に関するホーム・ページURLのデータベース475にアクセスすることができます。サーバは、データベース内の1企業の名前がHTMLページに現れるたびにハイパーテキスト・リンクを増加することを望んでいる。たとえば、HTMLページにIBMの名前が現れるたびに、サーバはIBMのホーム・ページへのハイパーテキスト・リンクの参照を変換することを望んでいるはずである。これを実行すれば、クライアントは、ポイントしクリックすることにより、会話中に現れる企業に関する有用な情報を得ることができるはずである。これは、データベースに現れるすべての企業名についてHTMLページを探索するように変換プログラム416を変更することによって達成することができる。このような名前が見つかると、その企業のホーム・ページへのハイパーテキスト・リンクがHTMLテキストに挿入され、クライアントに返される。変換プログラム416は、遠隔サーバへのハイパーテキスト・リンクをたどる場合、会話の監視を続行することができる。上記のように、クライアントがJavaなどの

言語で作成されたプログラムをサーバからダウンロードできる場合、変換プログラムはクライアント上で実行することができる。

【0082】代替実施例を備えた好ましい実施例として本発明を説明してきたが、当業者には様々な改良例が思い浮かぶだろう。したがって、好ましい実施例は例として示したものであって、限定として示したものではないことに留意されたい。本発明の範囲は請求の範囲によって適切に定義されている。

10 【0083】まとめとして、本発明の構成に関して以下の事項を開示する。

【0084】(1) 無状態プロトコルによりクライアントに対してネットワーク化された1つまたは複数のサーバからサービスを要求できるようになっているクライアントとの間の会話で状態情報を保存するためのコンピュータ化した方法において、前記サービスがクライアントが要求可能なデータおよびプログラムのうちの1つまたは複数を含み、会話は前記サービスに関するクライアントと1つまたは複数のサーバとの間の一連の通信であり、サーバからの各応答は前記サービスに関する別の要求を可能にする1つまたは複数の連結を含み、クライアントは会話を続行するためにその連結のうちの1つを呼び出さなければならず、前記方法が、クライアントが無状態プロトコルを使用してサーバとの会話を開始するステップと、サービス要求が状態情報の保存を必要とする時期を検出するステップと、前記検出ステップに応答して、前記サービスを実行し、前記サービスからの出力に含まれるすべての連結を識別するステップと、すべての識別済み連結に再帰的に状態情報を組み込むステップと、前記組込みステップに応答して、出力をクライアントに通信するステップとを含み、会話の持続期間中に状態情報が保存され、すべてのサービスに提供されることを特徴とする方法。

(2) 前記組込みステップがサーバによって実行され、前記通信ステップが前記組込みステップに応答して行われることを特徴とする、前記(1)に記載の方法。

(3) サーバに結合されたメモリに状態情報の少なくとも一部を格納するステップをさらに含み、前記組込みステップが前記すべての識別済み連結に状態情報の前記一部を表すインデックスを組み込むステップを含むことを特徴とする、前記(2)に記載の方法。

(4) 出力をクライアントに通信する前記ステップに応答して前記組込みステップを実行するために、コンピュータ・プログラム・コードをクライアントに動的にダウンロードするステップをさらに含むことを特徴とする、前記(1)に記載の方法。

(5) クライアントに結合されたメモリに状態情報の少なくとも一部を格納するステップをさらに含み、前記組込みステップが状態情報の前記一部を表すインデックスを組み込むステップを含むことを特徴とする、前記

(4) に記載の方法。

(6) クライアントが、前記すべての識別済み連結から状態情報が組み込まれた第2の連結を選択するステップと、前記第2の連結から状態情報を復元し、復元した状態情報により関連の第2のサービスを呼び出すステップと、前記第2のサービスからの出力に関するすべての連結において再帰的に状態情報を識別し、組み込むステップとをさらに含むことを特徴とする、前記(1)に記載の方法。

(7) 状態情報と特定の会話を相関させるステップをさらに含むことを特徴とする、前記(1)に記載の方法。

(8) クライアントとサーバがワールド・ワイド・ウェブを介してネットワーク化され、無状態プロトコルがハイパーテキスト転送プロトコルであり、連結が、ハイパーテキスト・マークアップ言語ファイルおよび共通ゲートウェイ・インターフェース・プログラムの1つへのハイパーリンクであることを特徴とする、前記(1)に記載の方法。

(9) 所定の基準に応じて前記サービスからのデータ出力と前記ハイパーリンクの1つをフィルタ処理するステップをさらに含むことを特徴とする、前記(8)に記載の方法。

(10) 所定の基準に応じて前記サービスからの前記出力にデータと前記ハイパーリンクの1つを追加するステップをさらに含むことを特徴とする、前記(8)に記載の方法。

(11) 前記組込みステップが、識別済み連結と引数として渡された状態情報によりCGI変換プログラムを呼び出すためにHTMLファイルを求める要求である識別済み連結を変更するステップをさらに含むことを特徴とする、前記(8)に記載の方法。

(12) 前記組込みステップが、識別済み連結と引数として渡された状態情報によるCGIプログラムへの呼出しである識別済み連結を変更するステップをさらに含み、前記組込みステップがCGIプログラムによって実行されることを特徴とする、前記(8)に記載の方法。

(13) 前記組込みステップが、識別済み連結によりCGI変換プログラムを呼び出すためのCGIプログラムへの呼出しである識別済み連結と、CGIプログラムに関連する引数の数を示す引数カウンタと、引数として渡された状態情報を変更するステップをさらに含み、前記組込みステップが変換プログラムによって実行されることを特徴とする、前記(8)に記載の方法。

(14) 無状態プロトコルによりクライアントに対してネットワーク化された1つまたは複数のサーバからサービスを要求できるようになっているクライアントとの間の会話で状態情報を保存するための方法を提供するためにコンピュータによって実行可能な複数の命令からなるプログラムを具体的に実施するコンピュータによって読み取るための方法。

取り可能なプログラム記憶装置において、前記サービスがクライアントが要求可能なデータおよびプログラムのうちの1つまたは複数を含み、会話は前記サービスに関するクライアントと1つまたは複数のサーバとの間の一連の通信であり、サーバからの各応答は前記サービスに関する別の要求を可能にする1つまたは複数の連結を含み、クライアントは会話を続行するためにその連結のうちの1つを呼び出さなければならず、前記方法が、クライアントが無状態プロトコルを使用してサーバとの会話を開始するステップと、サービス要求が状態情報の保存を必要とする時期を検出するステップと、前記検出ステップに応答して、前記サービスを実行し、前記サービスからの出力に含まれるすべてのハイパーリンクを識別するステップと、すべての識別済みハイパーリンクに再帰的に状態情報を組み込むステップと、前記組込みステップに応答して、出力をクライアントに通信するステップとを含み、会話の持続期間中に状態情報が保存され、すべてのサービスに提供されることを特徴とするプログラム記憶装置。

20 (15) 前記組込みステップがサーバによって実行され、前記通信ステップが前記組込みステップに応答して行われることを特徴とする、前記(14)に記載の方法ステップを実行するためにコンピュータによって実行可能な複数の命令からなるプログラムを具体的に実施するコンピュータによって読み取る可能なプログラム記憶装置。

(16) サーバに結合されたメモリに状態情報の少なくとも一部を格納するステップをさらに含み、前記組込みステップが前記すべての識別済み連結に状態情報の前記一部を表すインデックスを組み込むステップを含むことを特徴とする、前記(15)に記載の方法ステップを実行するためにコンピュータによって実行可能な複数の命令からなるプログラムを具体的に実施するコンピュータによって読み取る可能なプログラム記憶装置。

(17) 出力をクライアントに通信する前記ステップに応答して前記組込みステップを実行するために、コンピュータ・プログラム・コードをクライアントに動的にダウンロードするステップをさらに含むことを特徴とする、前記(14)に記載の方法ステップを実行するため40にコンピュータによって実行可能な複数の命令からなるプログラムを具体的に実施するコンピュータによって読み取る可能なプログラム記憶装置。

(18) クライアントに結合されたメモリに状態情報の少なくとも一部を格納するステップをさらに含み、前記組込みステップが状態情報の前記一部を表すインデックスを組み込むステップを含むことを特徴とする、前記

(14)に記載の方法ステップを実行するためにコンピュータによって実行可能な複数の命令からなるプログラムを具体的に実施するコンピュータによって読み取る可能なプログラム記憶装置。

(19) クライアントが、前記すべての識別済み連結から状態情報が組み込まれた第2の連結を選択するステップと、前記第2の連結から状態情報を復元し、復元した状態情報により関連の第2のサービスを呼び出すステップと、前記第2のサービスからの出力に関連するすべての連結において再帰的に状態情報を識別し、組み込むステップとをさらに含むことを特徴とする、前記(14)に記載の方法ステップを実行するためにコンピュータによって実行可能な複数の命令からなるプログラムを具体的に実施するコンピュータによって読み取り可能なプログラム記憶装置。

(20) 状態情報と特定の会話を相関させるステップをさらに含むことを特徴とする、前記(14)に記載の方法ステップを実行するためにコンピュータによって実行可能な複数の命令からなるプログラムを具体的に実施するコンピュータによって読み取り可能なプログラム記憶装置。

(21) クライアントとサーバがワールド・ワイド・ウェブを介してネットワーク化され、無状態プロトコルがハイパーテキスト転送プロトコルであり、連結が、ハイパーテキスト・マークアップ言語ファイルおよび共通ゲートウェイ・インターフェース・プログラムの1つへのハイパーリンクであることを特徴とする、前記(14)に記載の方法ステップを実行するためにコンピュータによって実行可能な複数の命令からなるプログラムを具体的に実施するコンピュータによって読み取り可能なプログラム記憶装置。

(22) 所定の基準に応じて前記サービスからのデータ出力と前記ハイパーリンクの1つをフィルタ処理するステップをさらに含むことを特徴とする、前記(21)に記載の方法ステップを実行するためにコンピュータによって実行可能な複数の命令からなるプログラムを具体的に実施するコンピュータによって読み取り可能なプログラム記憶装置。

(23) 所定の基準に応じて前記サービスからの前記出力にデータと前記ハイパーリンクの1つを追加するステップをさらに含むことを特徴とする、前記(21)に記載の方法ステップを実行するためにコンピュータによって実行可能な複数の命令からなるプログラムを具体的に実施するコンピュータによって読み取り可能なプログラム記憶装置。

(24) 前記組込みステップが、識別済み連結と引数として渡された状態情報によりCGI変換プログラムを呼び出すためにHTMLファイルを求める要求である識別済み連結を変更するステップをさらに含むことを特徴とする、前記(21)に記載の方法ステップを実行するためにコンピュータによって実行可能な複数の命令からなるプログラムを具体的に実施するコンピュータによって読み取り可能なプログラム記憶装置。

(25) 前記組込みステップが、識別済み連結と引数と

して渡された状態情報によるCGIプログラムへの呼出しである識別済み連結を変更するステップをさらに含み、前記組込みステップがCGIプログラムによって実行されることを特徴とする、前記(21)に記載の方法ステップを実行するためにコンピュータによって実行可能な複数の命令からなるプログラムを具体的に実施するコンピュータによって読み取り可能なプログラム記憶装置。

(26) 前記組込みステップが、識別済み連結によりCGI変換プログラムを呼び出すためのCGIプログラムへの呼出しである識別済み連結と、CGIプログラムに関連する引数の数を示す引数カウンタと、引数として渡された状態情報を変更するステップをさらに含み、前記組込みステップが変換プログラムによって実行されることを特徴とする、前記(21)に記載の方法ステップを実行するためにコンピュータによって実行可能な複数の命令からなるプログラムを具体的に実施するコンピュータによって読み取り可能なプログラム記憶装置。

(27) 無状態プロトコルによりクライアントに対してネットワーク化された1つまたは複数のサーバからサービスを要求できるようになっているクライアントとの間の会話を状態情報を保存するためのコンピュータ・システムにおいて、前記サービスがクライアントが要求可能なデータおよびプログラムのうちの1つまたは複数を含み、会話は前記サービスに関するクライアントと1つまたは複数のサーバとの間の一連の通信であり、サーバからの各応答は前記サービスに関する別の要求を可能にする1つまたは複数の連結を含み、クライアントは会話を続行するためにその連結のうちの1つを呼び出さなければならず、前記システムが、無状態プロトコルを使用してサーバとの会話を開始できるようになっているクライアントと、サービス要求が状態情報の保存を必要とする時期を検出する状態検出論理と、前記検出ステップに応答して、前記サービスからの出力に含まれるすべての連結を識別する探索論理と、すべての識別済み連結に再帰的に状態情報を組み込む変換プログラム論理と、出力をクライアントに通信する通信論理とを含み、会話の持続期間中に状態情報が保存され、すべてのサービスに提供されることを特徴とするコンピュータ・システム。

(28) 前記変換プログラム論理がサーバによって実行され、前記通信論理により状態情報が組み込まれた出力がサーバからクライアントに通信されることを特徴とする、前記(27)に記載のコンピュータ・システム。

(29) 状態情報の少なくとも一部を格納するためにサーバに結合されたメモリをさらに含み、前記変換プログラム論理が前記すべての識別済み連結に状態情報の前記一部を表すインデックスを組み込めるようになっていることを特徴とする、前記(28)に記載のコンピュータ・システム。

(30) 前記通信論理により状態情報が組み込まれてい

ない出力がサーバからクライアントに通信され、サーバが実行のために前記変換プログラム論理をクライアントに動的にダウンロードできるようになっていることを特徴とする、前記(27)に記載のコンピュータ・システム。

(31) 状態情報の少なくとも一部を格納するためにクライアントに結合されたメモリをさらに含み、前記変換プログラム論理がさらに状態情報の前記一部を表すインデックスを組み込めるようになっていることを特徴とする、前記(30)に記載のコンピュータ・システム。

(32) クライアントが、前記すべての識別済み連結から状態情報が組み込まれた第2の連結を選択し、変換プログラム論理が、さらに前記第2の連結から状態情報を復元し、復元した状態情報により関連の第2のサービスを呼び出し、前記第2のサービスからの出力に関連するすべての連結において再帰的に状態情報を識別し、組み込めるようになっていることをさらに含むことを特徴とする、前記(27)に記載のコンピュータ・システム。

(33) 状態情報と特定の会話とが相関することを特徴とする、前記(27)に記載のコンピュータ・システム。

(34) クライアントとサーバがワールド・ワイド・ウェブを介してネットワーク化され、無状態プロトコルがハイパーテキスト転送プロトコルであり、連結が、ハイパーテキスト・マークアップ言語ファイルおよび共通ゲートウェイ・インターフェース・プログラムの1つへのハイバーリンクであることを特徴とする、前記(27)に記載のコンピュータ・システム。

(35) 所定の基準に応じて前記サービスからのデータ出力と前記ハイバーリンクの1つをフィルタ処理するフィルタ論理をさらに含むことを特徴とする、前記(34)に記載のコンピュータ・システム。

(36) 所定の基準に応じて前記サービスからの前記出力にデータと前記ハイバーリンクの1つを追加する統合論理をさらに含むことを特徴とする、前記(34)に記載のコンピュータ・システム。

(37) 前記組込みステップが、識別済み連結と引数として渡された状態情報によりCGI変換プログラムを呼び出すためにHTMLファイルを求める要求である識別済み連結を変更するステップをさらに含むことを特徴とする、前記(34)に記載のコンピュータ・システム。

(38) 前記組込みステップが、識別済み連結と引数として渡された状態情報によるCGIプログラムへの呼出しである識別済み連結を変更するステップをさらに含み、前記組込みステップがCGIプログラムによって実行されることを特徴とする、前記(34)に記載のコンピュータ・システム。

(39) 前記組込みステップが、識別済み連結によりCGI変換プログラムを呼び出すためのCGIプログラムへの呼出しである識別済み連結と、CGIプログラムに

関連する引数の数を示す引数カウンタと、引数として渡された状態情報とを変更するステップをさらに含み、前記組込みステップが変換プログラムによって実行されることを特徴とする、前記(34)に記載のコンピュータ・システム。

(40) 無状態プロトコルによりクライアントに対してネットワーク化された1つまたは複数のサーバからサービスを要求できるようになっているクライアントとの間の会話で状態情報を保存するためのコンピュータ・システムにおいて、前記サービスがクライアントが要求可能なデータおよびプログラムのうちの1つまたは複数を含み、会話は前記サービスに関するクライアントと1つまたは複数のサーバとの間の一連の通信であり、サーバからの各応答は前記サービスに関する別の要求を可能にする1つまたは複数の連結を含み、クライアントは会話を続行するためにその連結のうちの1つを呼び出さなければならず、前記システムが、無状態プロトコルを使用してサーバとの会話を開始できるようになっているクライアントと、サービス要求が状態情報の保存を必要とする

20 時期を検出する状態検出手段と、前記検出手段に応答して、前記サービスからの出力に含まれるすべての連結を識別する探索手段と、すべての識別済み連結に再帰的に状態情報を組み込む変換プログラム手段と、出力をクライアントに通信する通信手段とを含み、会話の持続期間中に状態情報が保存され、すべてのサービスに提供されることを特徴とするコンピュータ・システム。

(41) 前記変換プログラム手段がサーバによって実行され、前記通信手段により状態情報が組み込まれた出力がサーバからクライアントに通信されることを特徴とする、前記(40)に記載のコンピュータ・システム。

(42) 状態情報の少なくとも一部を格納するためにサーバに結合されたメモリをさらに含み、前記変換プログラム手段が前記すべての識別済み連結に状態情報の前記一部を表すインデックスを組み込めるようになっていることを特徴とする、前記(41)に記載のコンピュータ・システム。

(43) 前記通信手段により状態情報が組み込まれていない出力がサーバからクライアントに通信され、サーバが実行のために前記変換プログラム手段をクライアント40 に動的にダウンロードできるようになっていることを特徴とする、前記(40)に記載のコンピュータ・システム。

(44) 状態情報の少なくとも一部を格納するためにクライアントに結合されたメモリをさらに含み、前記変換プログラム手段がさらに状態情報の前記一部を表すインデックスを組み込めるようになっていることを特徴とする、前記(43)に記載のコンピュータ・システム。

(45) クライアントが、前記すべての識別済み連結から状態情報が組み込まれた第2の連結を選択し、変換プログラム手段が、さらに前記第2の連結から状態情報を

39

復元し、復元した状態情報により関連の第2のサービスを呼び出し、前記第2のサービスからの出力に関連するすべての連結において再帰的に状態情報を識別し、組み込めるようになっていることをさらに含むことを特徴とする、前記(41)に記載のコンピュータ・システム。

(46) 所定の基準に応じて前記サービスからの前記出力にデータと前記ハイパーリンクの1つを追加する統合手段をさらに含むことを特徴とする、前記(45)に記載のコンピュータ・システム。

(47) クライアントとサーバがワールド・ワイド・ウェブを介してネットワーク化され、無状態プロトコルがハイパーテキスト転送プロトコルであり、連結が、ハイパーテキスト・マークアップ言語ファイルおよび共通ゲートウェイ・インターフェース・プログラムの1つへのハイパーリンクであることを特徴とする、前記(40)に記載のコンピュータ・システム。

(48) 前記組込みステップが、識別済み連結と引数として渡された状態情報によりCGI変換プログラムを呼び出すためにHTMLファイルを求める要求である識別済み連結を変更するステップをさらに含むことを特徴とする、前記(47)に記載のコンピュータ・システム。

(49) 前記組込みステップが、識別済み連結と引数として渡された状態情報によるCGIプログラムへの呼び出しである識別済み連結を変更するステップをさらに含み、前記組込みステップがCGIプログラムによって実行されることを特徴とする、前記(47)に記載のコンピュータ・システム。

(50) 前記組込みステップが、識別済み連結によりCGI変換プログラムを呼び出すためのCGIプログラムへの呼び出しである識別済み連結と、CGIプログラムに関連する引数の数を示す引数カウンタと、引数として渡された状態情報を変更するステップをさらに含み、前記組込みステップが変換プログラムによって実行されることを特徴とする、前記(47)に記載のコンピュータ・システム。

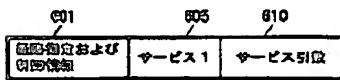
【図面の簡単な説明】

【図1】標準のブラウザによって表示されるHTMLページの一例を示す図である。

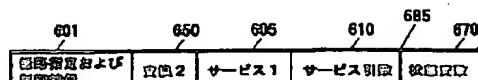
【図2】ウェブ・ブラウザによって表示されるHTMLの「用紙」の一例を示す図である。

【図3】「用紙」を使用して状態変数を保存するクライ

【図5】



【図6】



40

アントとサーバのブロック図である。

【図4】無状態プロトコルを使用してネットワーク上で本発明により状態を保存するための一般的な方法のブロック図である。

【図5】無状態プロトコルによる伝送用のデータ・パケットの汎用図である。

【図6】状態情報を保存するために本発明により変更された図5のデータ・パケットの一例を示す図である。

【図7】状態を保存しながら、ワールド・ワイド・ウェブ上でビジネス・トランザクションを行うための本発明の機能を有するシステムの一実施例を示す図である。

【図8】図7のシステム上で状態を保存するための方法の一実施例を示す図である。

【図9】図7に示すシステムおよび図8の方法ステップ745で状態を保存するための本発明による方法のより詳細な例を示す図である。

【図10】HTMLファイルへのハイパーテキスト・リンクの構造を示す図である。

【図11】本発明により状態を保存するために変更された図10の構造を示す図である。

【図12】タイプ1のCGIプログラムへのハイパーテキスト・リンクの構造を示す図である。

【図13】本発明による組込み状態引数を備えた図12の構造を示す図である。

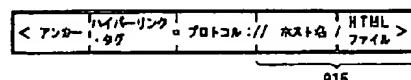
【図14】タイプ2のCGIプログラムへのハイパーテキスト・リンクの構造を示す図である。

【図15】本発明により状態を保存するために変更された図14の構造を示す図である。

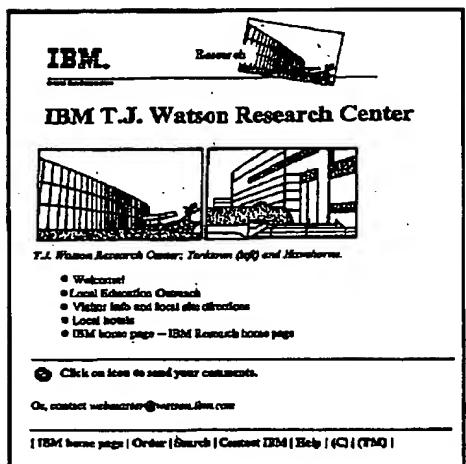
【符号の説明】

- 30 410' サーバ
- 412 DBMS
- 413 HTML/CGI
- 414 その他のCGIプログラム
- 415 「CGI P1」
- 416 変換プログラム
- 427 HTMLファイル
- 429 CGIプログラム
- 450 クライアント
- 460 ウェブ・ブラウザ
- 40 470 DASD
- 475 データベース

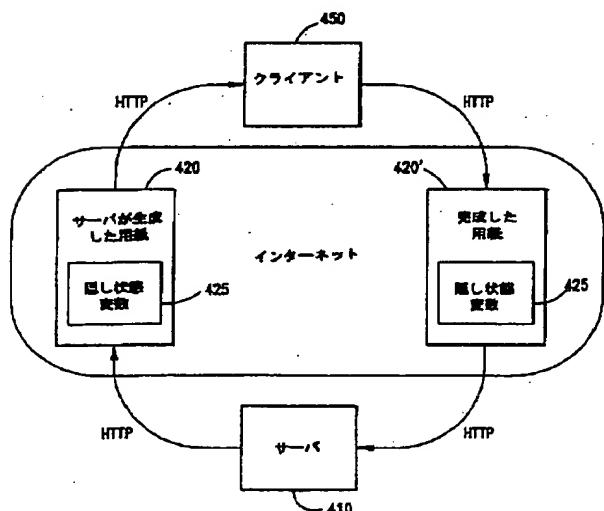
【図10】



【図1】



【図3】



【図11】

```
<アンカー>[ハイパーリンク]タグ プロトコル://ホスト名/[CGI実行!/?引数]ホスト名/HTML[状態]ファイル[実数]>
          820           915           930
```

【図13】

```
<アンカー>[ハイパーリンク]タグ プロトコル://ホスト名/[CGI実行!/?引数]>[状態]文件[実数]>
          940           950           960
```

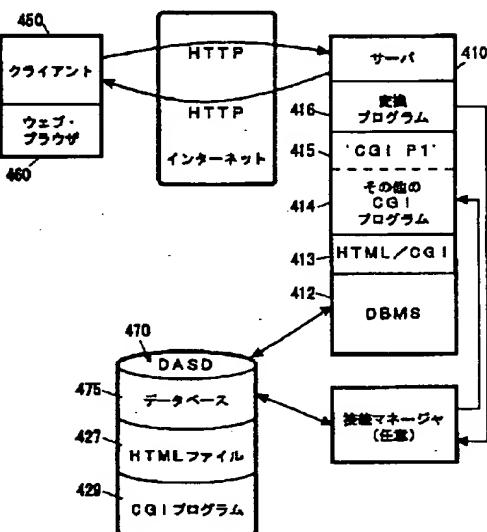
【図2】

SKYLINE SUPPLIES REGISTRATION FORM

USERID	[]
PASSWORD	[]
PASSWORD (FOR VERIFICATION)	[]
ACTUAL NAME	[]
COMPANY	[]
E-MAIL ADDRESS	[]
PHONE NUMBER	[]

SEND RESET (RESET FORM)

【図7】



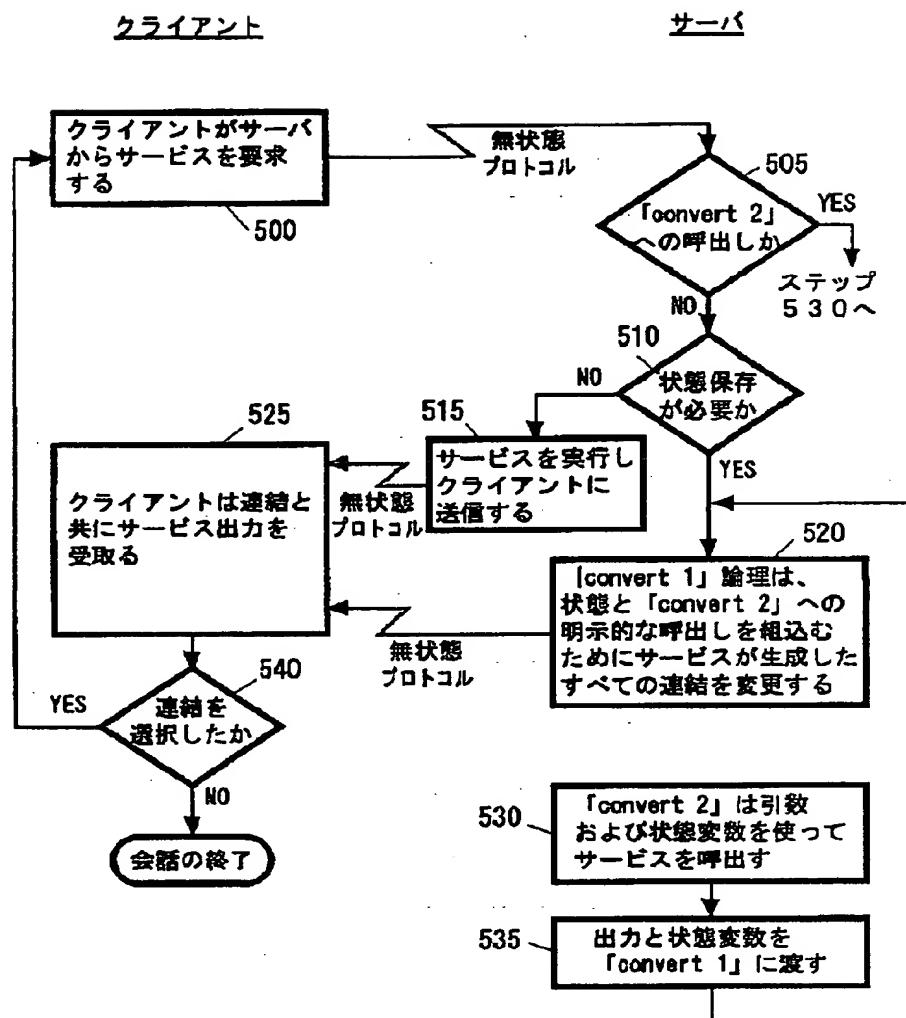
【図12】

```
<アンカー>[ハイパーリンク]タグ プロトコル://ホスト名/[CGI実行!/?引数]>[状態]ファイル[実数]>
          940
```

【図14】

```
<アンカー>[ハイパーリンク]タグ プロトコル://ホスト名/[CGI実行!/?引数]>[状態]サービス[実数]>
          950           960
```

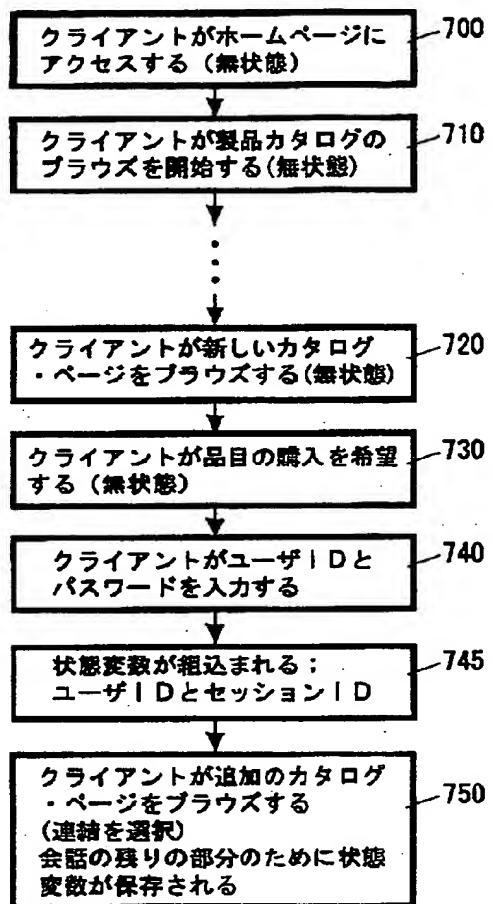
【図4】



【図15】

ハイパーリンク	プロトコル : // ホスト名 / 「実動プログラム」	CGI呼び出し	数値 引数	状態 変数
<アンカ>・タグ			920° 950° 970° 960° 930°	

【図8】



【図9】

